

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

#### Extraction de données touristiques et visualisation sur PDA

Maurice, Cyrille

*Award date:*  
2007

*Awarding institution:*  
Université de Namur

[Link to publication](#)

#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

**EXTRACTION DE DONNÉES  
TOURISTIQUES  
ET VISUALISATION SUR PDA**

*Cyrille Maurice*

---

Mémoire présenté en vue de l'obtention  
du grade de Maître en Informatique

**Année académique 2006-2007**







## Résumé

Avec l'émergence du Web, les outils actuels de recherches ne sont plus adaptés à une masse considérable d'informations et les tentatives de « triche » de certains afin d'augmenter le trafic vers leurs sites. Résultat : les résultats retournés par les moteurs de recherches classiques (comme Google, MSN ou bien Yahoo) ne sont pas toujours de qualité par rapport aux attentes des utilisateurs... Pour contrer ces phénomènes et avec la naissance du Web 2.0, on essaye de voir ce que pourrait apporter la sémantique aux moteurs de recherches actuels et principalement au niveau de l'adéquation des résultats par rapport à la requête de l'utilisateur. Nous nous rendons compte que de plus en plus, nous utilisons des technologies comme les ontologies ou les thésaurus pour améliorer la gestion de l'information et relier le contenu entre eux.

Ainsi, dans un premier temps, nous définirons le web sémantique et les différents éléments qui le composent comme RDF, les Topic Maps, les ontologies, etc. Nous en profiterons pour également présenter plusieurs solutions concernant la visualisation de systèmes complexes, sémantiquement structurés.

C'est dans cette perspective que le projet Eiffel tente de développer un portail sémantique pour le tourisme avec en plus un système d'analyse du comportement de l'utilisateur basé sur l'analyse des logs afin de personnaliser l'affichage du site. La plateforme utilisée pour ce portail utilise des ontologies et des thésaurus traités précédemment.

Avant d'analyser le domaine touristique existant dans le but de nous aider à définir le contenu des fiches touristiques que nous souhaitons mettre en place, nous nous intéresserons à plusieurs thématiques : la personnalisation et la localisation d'applications, la nouvelle génération de moteurs de recherches basés sur la sémantique et enfin les caractéristiques et contraintes du développement pour les périphériques mobiles.

Par la suite, nous étudierons, dans ce mémoire, la manière d'adapter le contenu et plus particulièrement l'affichage et la visualisation des fiches touristiques et des requêtes de recherches aux périphériques mobiles. Afin de proposer un layout accessible à tous (même aux personnes handicapées), nous parcourrons les recommandations du W3C et d'autres organismes spécialisés dans le domaine de l'accessibilité du web classique et mobile. Nous développerons ensuite une structure sémantique pour nos fiches pour terminer par proposer des méthodes de visualisation du contenu adaptées aux petits écrans en essayant d'innover par rapport à ce qui existe actuellement dans le domaine touristique, tout support confondu.

## Abstract

With the emergence of the web, the present tools of search are not adapted to the increasing of information and the "hack" to increase the number of visitors. So the results given by the classic search engines (like Google, MSN or Yahoo) are not always what the user want... So to counter these phenomena and with the coming of the Web 2.0, we try to see what the semantic could bring to improve the adequacy between the request of the user and the results returned.

We use more and more often technologies like the ontologies or the thesaurus to improve the management of information and to link content between them. In a first time we will explain what the semantic web is and several elements form it like RDF, Topic Maps, ontologies, ... We will also present some solutions about visualization of complex systems semantically structured.

It is in this perspective that the Eiffel's project tries to develop a semantic portal for the tourism with the personalization of interfaces according to the behavior of the user. The platform used for the portal is based on ontologies and thesaurus to refine the requests of users and improve the results.

Before analyzing the touristy domain already existing to help us to define the content of the touristy cards, we will talk about several thematic : the personalization and the localisation of softwares, the "new generation" of search engines based on the semantic and the characteristics and constraints of the development for mobile devices.

Next we will search how to adapt the content and more particularly the visualization of the touristy cards and the requests of users for mobile devices. We will skim the guidelines of the W3C and of others organisms specialized in the web's accessibility.

Finally we will define a semantic structure for the cards and we will propose some methods to adapt the content to the little screen of PDAs and smartphones. In the touristy domain, we will try to get what already exists better.

## Remerciements

Dans un premier temps, je tenais à remercier différentes personnes sans qui ce travail n'aurait pas pu être fait :

Madame Monique Noirhomme-Fraiture, promoteur de ce mémoire pour son encadrement et ses précieux conseils pour la rédaction de ce présent mémoire ;

Monsieur Yves Lechevallier, directeur de recherche à l'INRIA et vice-directeur de l'équipe de projet AxIS pour son accueil au sein de l'INRIA de Rocquencourt et m'avoir permis de réaliser mon stage dans un cadre scientifique réputé et m'avoir également encadré dans mon travail en tant que maître de stage avec Marie-Aude Aufaure, partenaire scientifique du département informatique de Supélec à Gif-sur-Yvette avec l'INRIA ;

L'ensemble des chercheurs, doctorants ou post-doctorant et membres de l'équipe d'AxIS : Brigitte Trousse, Thierry Despeyroux, Fabrice Rossi, Anne-Marie Vercoustre, Abdourahmane Balde, Alzennyr da Silva, Zeina Jrad et Jovan Pehcevski pour leur accueil, leur aide et leurs réponses à mes questions tout au long de mon stage ;

Un remerciement particulier à Stéphanie Aubin, assistante de l'équipe d'AxIS de Rocquencourt pour son aide dans toutes les démarches administratives pour mon stage et mon intégration au sein de l'INRIA ;

L'ensemble de ma famille et principalement mes parents pour leur soutien moral et financier tout au long de mes études et plus particulièrement depuis que je suis en Belgique, Catherine pour m'avoir aidé et conseillé pour certaines choses dans mon mémoire et sa relecture.





---

## Table des matières

---

<b>Glossaire</b>	<b>1</b>
<b>1 Introduction</b>	<b>3</b>
1.1 Présentation du projet Eiffel . . . . .	3
1.2 Problématique du mémoire . . . . .	5
<b>2 Notions introductives</b>	<b>7</b>
2.1 Web Sémantique . . . . .	7
2.1.1 Introduction . . . . .	7
2.1.2 Objectifs du Web Sémantique . . . . .	7
2.1.3 Architecture du Web Sémantique . . . . .	8
2.1.4 RDF . . . . .	9
2.1.5 Topic Maps . . . . .	11
2.1.6 Ontologie . . . . .	12
2.1.7 Visualisation de systèmes complexes sémantiquement structuré . . . . .	14
Ontopia . . . . .	14
TMNav . . . . .	15
Autres pistes de recherches . . . . .	17
<b>3 Recherches préliminaires</b>	<b>19</b>
3.1 Personnalisation/adaptation . . . . .	19
3.1.1 Personnalisation/adaptation d’interfaces/applications . . . . .	19
3.1.2 Localisation . . . . .	22
3.2 Nouvelle génération de moteur de recherche . . . . .	24
3.3 Caractéristiques et contraintes de développement du monde des périphériques mobiles . .	27
<b>4 Présentation du domaine touristique et de l’existant</b>	<b>31</b>
4.1 Analyse du site de la Nièvre . . . . .	31
4.2 Analyse du domaine du tourisme . . . . .	34
4.2.1 Le tourisme sur Internet . . . . .	34
ViaMichelin . . . . .	34
Site de l’office du tourisme de la Nouvelle-Zélande . . . . .	34
MadeinMtl . . . . .	35
4.2.2 Les guides touristiques . . . . .	35
4.2.3 Autres supports . . . . .	36

4.2.4	Conclusion . . . . .	36
<b>5</b>	<b>Analyse et structuration des fiches touristiques</b>	<b>39</b>
5.1	Structure des fiches touristiques existantes . . . . .	39
5.1.1	Contenu des fiches du site de la Nièvre . . . . .	39
5.1.2	Contenu des fiches du site Via Michelin . . . . .	39
5.2	Réflexions sur le contenu des fiches touristiques . . . . .	41
5.2.1	Suggestions supplémentaires . . . . .	41
5.3	Schéma entité-relation des fiches . . . . .	42
5.4	Stockage physique du contenu des fiches . . . . .	45
5.5	Structure physique des fiches touristiques . . . . .	45
5.5.1	Analyse de la structure physique des fiches touristiques . . . . .	45
5.5.2	Sémantique des éléments de la DTD . . . . .	50
	Précisions sur les XML Schema . . . . .	54
5.5.3	Syntaxe des éléments de la DTD . . . . .	54
<b>6</b>	<b>Accessibilité et ergonomie dans les IHM</b>	<b>59</b>
6.1	Recommandations du W3C concernant l'accessibilité . . . . .	59
6.2	Évaluation automatique de l'ergonomie des site web . . . . .	61
6.3	Page imprimée vs page web . . . . .	61
6.4	Recommandations du W3C pour le web mobile . . . . .	62
6.4.1	Navigation et liens . . . . .	62
6.4.2	Mise en page et contenu . . . . .	63
6.4.3	Définition des pages . . . . .	64
6.4.4	Données en entrée . . . . .	64
6.5	Étude sur les requêtes écrites et orales pour l'accès à l'information dans un contexte mobile	64
6.5.1	Contexte de l'étude . . . . .	64
6.5.2	Présentation de l'expérience . . . . .	65
6.5.3	Résultats . . . . .	65
6.5.4	Conclusions . . . . .	67
6.6	Conclusion . . . . .	68
<b>7</b>	<b>Layout des fiches touristiques</b>	<b>69</b>
7.1	Idées générales de la présentation . . . . .	69
7.1.1	Propositions de layout . . . . .	69
7.1.2	Réflexions sur les rubriques des fiches . . . . .	71
7.2	Représentation graphique conceptuelle des interfaces . . . . .	73
7.2.1	Représentation graphique du nom . . . . .	74
7.2.2	Représentation graphique des coordonnées géographiques . . . . .	74
7.2.3	Représentation graphique des coordonnées . . . . .	74
7.2.4	Représentation graphique de la description . . . . .	75
7.2.5	Représentation graphique des photos/vidéos . . . . .	76
7.2.6	Représentation graphique des horaires et périodes d'ouverture . . . . .	77
7.2.7	Représentation graphique des prix, des tarifs des activités . . . . .	79
7.2.8	Représentation graphique des moyens de paiement . . . . .	79
7.2.9	Représentation graphique des services/équipements offerts . . . . .	80
7.2.10	Représentation graphique du type de standing . . . . .	81
7.2.11	Représentation graphique des distinctions, labels . . . . .	81
7.2.12	Représentation graphique du type de gastronomie . . . . .	81
7.2.13	Représentation graphique du public visé . . . . .	82
7.2.14	Représentation graphique de la durée de l'activité . . . . .	82
7.2.15	Représentation graphique du nombre de places totales disponibles . . . . .	83
7.2.16	Représentation graphique de la répartition des chambres par type . . . . .	84

7.2.17	Représentation graphique de la superficie des emplacements . . . . .	84
7.2.18	Représentation graphique de la note d'intérêt . . . . .	84
7.3	Représentation graphique des requêtes de recherche . . . . .	85
7.4	Affichage des résultats d'une recherche . . . . .	88
7.4.1	Représentation graphique du panier . . . . .	90
7.5	Implémentation possible des fiches touristiques . . . . .	91
<b>8</b>	<b>Conclusion</b>	<b>93</b>
<b>A</b>	<b>Code source du layout du site web</b>	<b>99</b>
A.1	Onglet Description . . . . .	99
A.1.1	Code HTML . . . . .	99
A.1.2	Code CSS . . . . .	100
A.2	Onglet Coordonnées . . . . .	102
A.2.1	Code HTML . . . . .	102
A.2.2	Code CSS . . . . .	103
A.3	Choix du type d'objet dans une recherche . . . . .	104
A.3.1	Code HTML . . . . .	104
A.3.2	Code CSS . . . . .	104
A.4	Composition de la famille dans une recherche . . . . .	105
A.4.1	Code HTML . . . . .	105
A.4.2	Code CSS . . . . .	106
A.4.3	Javascript . . . . .	107
<b>B</b>	<b>Code source du site de la Nièvre</b>	<b>109</b>
B.1	Script XSL d'affichage d'une fiche touristique . . . . .	109
B.2	Fichier XML d'une fiche touristique . . . . .	109
<b>C</b>	<b>Code source des exemples de fiches touristique</b>	<b>111</b>
C.1	Fiche d'une activité . . . . .	111
C.2	Fiche d'un restaurant . . . . .	115
C.3	Fiche d'un hébergement . . . . .	120



---

## Table des figures

---

1.1	Architecture fonctionnelle du projet Eiffel . . . . .	4
1.2	Détail des sous projets technologiques et de leurs dépendances . . . . .	4
2.1	Architecture du Web Sémantique . . . . .	8
2.2	Exemple d'un graphe RDF . . . . .	9
2.3	Représentation graphique de la déclaration RDF . . . . .	10
2.4	Exemple de Topic Map . . . . .	11
2.5	Illustration des composantes traditionnelles d'une ontologie . . . . .	13
2.6	Affichage d'un Topic Map avec Vizigation . . . . .	15
2.7	Affichage d'un Topic Map avec TMNav dans le mode « Touchgraph » . . . . .	16
2.8	Affichage d'un Topic Map avec TMNav dans le mode « Section » . . . . .	16
2.9	Menu de TMNav . . . . .	17
2.10	Métaphore de la ville virtuelle [14] . . . . .	18
2.11	City of News [48] . . . . .	18
3.1	Interface de la version classique [19] . . . . .	21
3.2	Interfaces de la version mobile [19] . . . . .	21
3.3	Un extrait de la liste de résultats proposée par le moteur de recherche Google en réponse à la requête « orange » (recherche réalisée sur le Web francophone) [14] . . . . .	24
3.4	Listes de résultats proposées par le moteur de recherche <i>Clusty</i> en réponse à la requête « orange » [14] . . . . .	25
3.5	KartOO, méta-moteur de recherche cartographique en 2D [14] . . . . .	26
3.6	MapStan Search, méta-moteur de recherche cartographique en 2D [14] . . . . .	27
3.7	Exemple sous Opera 8.65 Beta . . . . .	29
3.8	Exemple sous Minimo 0.2 . . . . .	29
4.1	Page d'accueil du site de la Nièvre . . . . .	32
4.2	Recherche en plein texte . . . . .	32
4.3	Recherche en utilisant l'ontologie . . . . .	32
4.4	Historique des étapes de la recherche . . . . .	32
4.5	Affichage de la fiche d'un objet . . . . .	33
4.6	Problème d'affichage des suggestions . . . . .	33
5.1	Version 1 du schéma ERA de la structure des fiches . . . . .	43
5.2	Version 2 du schéma ERA de la structure des fiches . . . . .	46

5.3	Version finale du schéma ERA de la structure des fiches . . . . .	51
6.1	Exemple d'un topic du TREC . . . . .	65
6.2	Caractéristiques des requêtes écrites . . . . .	66
6.3	Caractéristiques des requêtes orales . . . . .	66
6.4	Longueur moyenne des requêtes par topic . . . . .	66
6.5	Longueur moyenne des requêtes par utilisateur . . . . .	66
6.6	Longueur moyenne des requêtes par topic . . . . .	67
6.7	Longueur moyenne des requêtes par utilisateur . . . . .	67
7.1	Layout basé sur une barre de navigation . . . . .	70
7.2	Layout basé sur un frameset . . . . .	70
7.3	Thématiques des informations des fiches touristiques . . . . .	72
7.4	Exemple d'une fiche du site des gîtes de France . . . . .	73
7.5	Layout pour les coordonnées de l'objet . . . . .	75
7.6	Layout pour la présentation générale de l'objet . . . . .	76
7.7	Exemple de diaporama dynamique . . . . .	77
7.8	Système de disponibilité d'un hébergement des gîtes de France . . . . .	78
7.9	Layout pour les horaires et les périodes d'ouvertures . . . . .	78
7.10	Layout pour les prix . . . . .	79
7.11	Visa . . . . .	80
7.12	Mastercard . . . . .	80
7.13	Chèque . . . . .	80
7.14	Espèce . . . . .	80
7.15	Distances entre différentes activités et l'hébergement . . . . .	80
7.16	Services/équipements offerts . . . . .	80
7.17	Gaultmillau . . . . .	81
7.18	Gîtes de France . . . . .	81
7.19	Michelin . . . . .	81
7.20	Type de gastronomie . . . . .	81
7.21	Public visé . . . . .	82
7.22	Durée d'une activité . . . . .	83
7.23	Nombres de places totales disponibles . . . . .	83
7.24	Répartition des chambres par type . . . . .	84
7.25	Note d'intérêt . . . . .	85
7.26	Choix du type d'objet touristique recherché . . . . .	86
7.27	Classement de préférences de l'utilisateur par priorité . . . . .	87
7.28	Encodage de la composition de la famille . . . . .	87
7.29	Écran d'affichage des résultats d'une recherche . . . . .	89
7.30	Graphique en étoile . . . . .	90

## Glossaire

**Unicode** Unicode est une norme informatique, développée par le Consortium Unicode, qui vise à donner à tout caractère de n'importe quel système d'écriture de langue un nom et un identifiant numérique, et ce de manière unifiée, quelle que soit la plate-forme informatique ou le logiciel.<sup>1</sup>

**URI** URI pour Uniform Resource Identifier est un identifiant unique respectant la norme émise par le W3C pour les ressources du Web. L'identifiant est une chaîne de caractères. Voici un exemple d'URI : *http ://www.exemple.com/index.html* ou bien *ftp ://ftp.exemple.com/index.html*, etc.

**RDF** Resource Description Framework est un modèle de graphe destiné à décrire de façon formelle les ressources Web et leurs métadonnées, de façon à permettre le traitement automatique de telles descriptions. Développé par le W3C, RDF est le langage de base du Web sémantique. Une des syntaxes (sérialisation) de ce langage est RDF/XML.<sup>2</sup>

**Ontologie** En informatique, une ontologie est un système de représentation des connaissances.

**Topic Map** Les Topic Maps sont un outil très général de représentation des connaissances, dont le but est d'agréger autour d'un point unique d'indexation (appelé topic) toutes les informations disponibles concernant un sujet donné, et de relier ces points par un réseau sémantique de relations appelées associations.<sup>3</sup>

**DTD** Une DTD indique les noms des éléments pouvant apparaître et leur contenu, i.e. les sous-éléments et les attributs. En dehors des attributs, le contenu est spécifié en indiquant le nom, l'ordre et le nombre d'occurrences autorisées des sous-éléments. L'ensemble constitue la définition des hiérarchies valides d'éléments et de texte.<sup>4</sup>

**Thesaurus** Un thesaurus ou thésaurus est une sorte de dictionnaire hiérarchisé ; un vocabulaire normalisé sur la base de termes génériques et de termes spécifiques à un domaine. Il ne fournit qu'accessoirement des définitions, les relations des termes et leur choix l'emportant sur les significations.<sup>5</sup>

**RNTL** Le Réseau National de recherche et d'innovation en Technologies Logicielles (RNTL) a été mis en place, fin 1999, par le Ministère délégué à la Recherche et le Ministère délégué à l'Industrie. Le RNTL a été renouvelé en 2005, dans le cadre de l'Agence Nationale de la Recherche (ANR). Il a reçu comme mission de favoriser la constitution de projets innovants de recherche et de développement coopératif entre entreprises et équipes de la recherche publique. Ces actions sont ouvertes à toutes les entreprises (PME et grands groupes de tous les secteurs économiques) en coopération avec des équipes de la recherche publique dans le domaine du logiciel.

**MySQL** MySQL est un système de gestion de base de données libre. Il est très utilisé dans les projets libres et dans le milieu industriel, en particulier dans le domaine du web.

**Cluster** En statistique, le mot cluster désigne le regroupement d'individus en fonction de leur caractéristiques. Le but du clustering est d'obtenir des groupes composés d'individus les plus homogènes possibles et où les différents groupes sont les plus contrastés possibles.

---

<sup>1</sup>Définition issue de Wikipédia

<sup>2</sup>Définition issue de Wikipédia

<sup>3</sup>Définition issue de Wikipédia

<sup>4</sup>Définition issue de Wikipédia

<sup>5</sup>Définition issue de Wikipédia

**Web 2.0** Web 2.0 est un terme souvent utilisé pour désigner ce qui est perçu comme une transition importante du World Wide Web, passant d'une collection de sites Web à une plate-forme informatique à part entière, fournissant des applications Web aux utilisateurs. Les défenseurs de ce point de vue soutiennent que les services du Web 2.0 remplaceront progressivement les applications traditionnelles de bureau.<sup>6</sup>

**Service Web** Un service Web est un ensemble de protocoles et de normes informatiques utilisés pour échanger des données entre les applications. Afin de garantir l'interopérabilité entre les différentes plate-formes et applications, le W3C et l'OSI ont développé des standards pour ces protocoles et normes que l'on nomme SOA (Service Oriented Architecture).

**Ranking** Position dans le classement établie pour déterminer l'ordre d'affichage des résultats d'une recherche. La calcul du ranking n'est pas une chose aisée et est propre à chaque moteur de recherche (qui garde secret sa méthode de calcul pour éviter les tentatives de « fraudes ») mais il est basé en grande partie sur le nombre de liens dans l'ensemble du web qui pointe vers la page donnée.

**RSS** RSS désigne une famille de formats XML utilisés pour la syndication<sup>7</sup> de contenu Web. Ce système est habituellement utilisé pour diffuser les mises à jour de sites dont le contenu change fréquemment ce qui permet à l'utilisateur abonné à ce flux de consulter rapidement les dernières mises à jour sans avoir à se rendre sur le site.<sup>8</sup>

**Widget** Synonyme de gadget (basé sur la contraction de Windows Gadget). En informatique, il s'agit d'un programme situé sur le bureau et qui permet d'accomplir certaines tâches comme afficher les prévisions de la météo, les cours de la bourse, ...

**API** API ou *Application Programming Interface*, c'est-à-dire *Interface de programmation* en français, est une interface normalisée permettant à un logiciel de faire appel aux fonctions d'un autre programme déjà disponible sur une machine.<sup>9</sup>

**Podcast** Podcast est un mot formé sur la base iPod, le baladeur multimédia d'Apple et du mot anglais broadcast (diffusion). Le podcast est un fichier multimédia diffusé sur Internet et que l'on peut transférer sur un baladeur MP3 pour l'écouter quand on le souhaite.

**User Agent** Un user agent est un logiciel qui recherche, récupère et s'occupe du rendu d'un contenu internet comme une page d'un site web.

**CSS** Cascading Style Sheets en anglais ou bien feuilles de style (en cascade). Il s'agit d'un ensemble d'instructions permettant de mettre en forme et de positionner du contenu dans une page HTML ou bien XML. Les possibilités des CSS sont beaucoup plus évoluées et puissantes que les instructions que l'on retrouve dans le langage HTML.

---

<sup>6</sup>Définition issue de Wikipédia

<sup>7</sup>Publication ou diffusion d'un contenu à plusieurs endroits en même temps

<sup>8</sup>Définition issue de Wikipédia

<sup>9</sup>Définition du guide de l'informatique



### 1.1 Présentation du projet Eiffel

Le projet Eiffel est un projet RNTL dont l'objectif est de mettre en œuvre une plateforme « moteur de recherche tourisme/territoire » afin de valoriser l'offre touristique au niveau du territoire et de faciliter l'accessibilité et la compréhension de l'offre du côté des utilisateurs. Le projet souhaite utiliser le web sémantique afin d'améliorer la qualité des résultats des recherches des utilisateurs et également proposer de nouvelles méthodes de recherche tel que l'utilisation d'ontologie (que nous définirons dans la section 2.1.6 page 12), des thesaurus, ...

On souhaite également ajouter des services de personnalisation de l'application web en fonction des habitudes d'utilisation de l'utilisateur et offrir un service de « panier » où l'utilisateur pourra stocker les résultats de recherches qui l'intéressent afin de pouvoir les retrouver facilement lors d'une prochaine visite sur le site.

Du fait de la nature d'un projet RNTL, plusieurs partenaires, qu'ils soient privés ou publics, sont impliqués dans le projet Eiffel. Du côté des partenaires privés, on retrouve les sociétés Antidot<sup>1</sup> et Mondeca<sup>2</sup> et du côté public, on retrouve les centres de recherches de l'INRIA<sup>3</sup>, du LIRMM<sup>4</sup> et de Modyco<sup>5</sup>.

Les objectifs visés par le projet sont les suivants [10] : "Le projet Eiffel est l'occasion pour les deux éditeurs de prendre une avance technologique significative dans les solutions de moteur de recherche sémantique spécialisé tourisme en s'alliant aux meilleurs laboratoires de recherche sur trois domaines particulièrement sensibles que sont :

- l'annotation sémantique et l'acquisition automatique de connaissances à partir des technologies du Traitement Automatique du Langage ;
- le raisonnement automatisé ;
- les interfaces utilisateurs et les fonctions de requêtes sur des ontologies et du contenu distribué."

Les résultats attendus à la fin des 3 ans du projet sont les suivants :

- une plateforme technique de moteur de recherche web sémantique dans le domaine du e-tourisme commercialisée auprès des collectivités et promoteurs. La solution issue du projet est adaptée à la

---

<sup>1</sup>[www.antidot.net](http://www.antidot.net)

<sup>2</sup>[www.mondeca.com/fr/index.htm](http://www.mondeca.com/fr/index.htm)

<sup>3</sup>Institut National de Recherche en Informatique et en Automatique, [www.inria.fr](http://www.inria.fr)

<sup>4</sup>Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, [www.lirmm.fr](http://www.lirmm.fr)

<sup>5</sup>Laboratoire Modèles, Dynamiques, Corpus, [www.modyco.fr](http://www.modyco.fr)

commercialisation internationale d'une offre touristique de pays francophone. Des versions pour des territoires non francophones seront ensuite mises en place par les partenaires industriels ;

- un moteur de recherche tourisme/territoire généraliste exploité par Antidot/Mondeca faisant l'objet d'accord croisé avec les territoires ;
- une plateforme technique de moteur de recherche web sémantique pouvant être réutilisé pour d'autres applications de moteurs de recherche spécialisés par Antidot ou Mondeca (juridique, éducation, science...).

Voici l'architecture souhaitée pour le projet Eiffel [10] afin de se donner une idée des différentes tâches de recherches nécessaires :

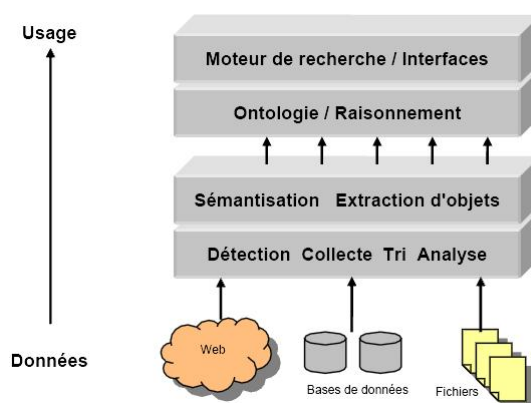


Figure 1.1 – Architecture fonctionnelle du projet Eiffel

Nous avons émis le choix de ne pas détailler cette architecture car il ne s'agit pas du thème de notre travail et puis surtout, l'INRIA n'est pas impliqué dans la totalité du processus du projet Eiffel. L'équipe Axis<sup>6</sup> est responsable du sous-projet 8 *Interfaces utilisateurs : recherche, navigation, sélection* qui sera détaillé par la suite. Afin d'avoir une vue globale du projet, voici le détail des sous projets technologiques et de leurs dépendances [10] :

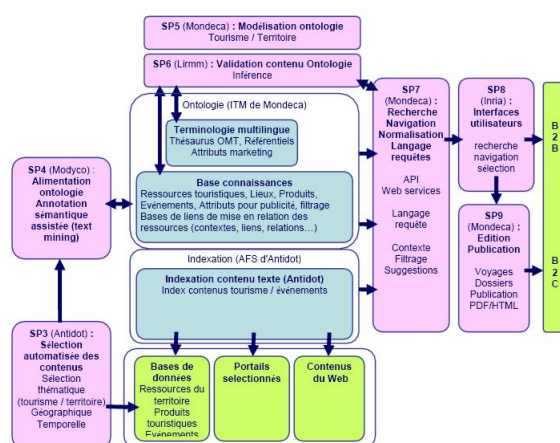


Figure 1.2 – Détail des sous projets technologiques et de leurs dépendances

<sup>6</sup><http://www-sop.inria.fr/axis/>

Maintenant, terminons la présentation du projet Eiffel en présentant les tâches incombant à l'INRIA, celles du sous-projet 8.

Comme on le voit sur la figure 1.2 page précédente, l'INRIA est en charge des interfaces utilisateurs mais également de la personnalisation des services offerts aux utilisateurs en fonction de son comportement de navigation qui n'est pas indiqué sur le schéma.

Concernant les interfaces utilisateurs, on souhaite avoir un site graphique, intuitif, simple d'utilisation et permettant aussi bien d'afficher les résultats que d'effectuer les requêtes et de sélectionner les résultats. De plus, on souhaite offrir des services nouveaux aux utilisateurs comme un « panier » afin qu'il puisse mémoriser les résultats qui l'intéresse pour les retrouver lors d'une prochaine visite, planifier son séjour, réserver son logement, etc...

L'autre tâche, qui est plus la spécialité de l'équipe Axis, consiste à analyser les logs des sites internet afin de dégager des profils d'utilisateurs (cluster) par des méthodes et algorithmes mathématiques du domaine du Data Mining. Dans ce but, l'INRIA a développé un outil, AxisLogMiner, qui permet de traiter les logs internet et de les transformer en base de données relationnelles type MySQL afin de pouvoir ensuite effectuer des requêtes et obtenir les informations souhaitées. Le but de cela dans le projet Eiffel est de pouvoir offrir des services personnalisés aux utilisateurs en fonction de leur profil de navigation (débutant, avancé, expert, ...) sans avoir besoin de lui demander ces informations lors de son inscription sur le site comme le font habituellement les sites web (loisirs, activité professionnelle, ...) mais en les « devinant » par la manière dont l'utilisateur navigue sur le site. Évidemment, ces profils évolueront au fur et à mesure des visites de l'utilisateur et ne seront pas statiques.

## 1.2 Problématique du mémoire

Comme nous l'avons vu précédemment, la grande part du travail attribuée à l'INRIA se situe au niveau des interfaces graphiques pour les utilisateurs. Dans le projet Eiffel, il n'est indiqué nulle part d'une adaptation aux périphériques mobiles tels que les smartphone, PDA, ... mais certains partenaires étaient intéressés par cette éventualité et nous avons donc été chargé d'effectuer des recherches et de voir la manière dont on pourrait adapter le projet pour ce genre de périphériques.

Étant donné que le projet en est à ses débuts, il n'existe pas de données structurées sur lesquelles se baser pour effectuer ce travail. Nous allons donc dans un premier temps définir et structurer précisément l'information dont nous allons avoir besoin afin, ensuite, de spécifier les interfaces graphiques pour les périphériques mobiles. La démarche que nous allons suivre dans ce mémoire est la suivante :

- définir les notions que nous utiliserons dans la suite du mémoire
- définir et structurer les données nécessaires pour les fiches touristiques
- définir une charte graphique pour le projet pour périphériques mobiles et spécifier les interfaces graphiques
- expliquer la manière dont on pourrait développer et implémenter les spécifications
- évolution future possible et/ou pistes pour poursuivre le travail dans ce domaine

Il est maintenant temps de définir les concepts et notions que nous utiliserons par la suite dans ce mémoire.



## 2.1 Web Sémantique

### 2.1.1 Introduction

À une époque où Internet se développe de plus en plus, le soucis n'est plus l'accès à l'information comme il y a plusieurs centaines d'années, avant le développement de l'imprimerie où seul les érudits pouvaient être instruits, mais de retrouver l'information. En effet, que ce soit pour les entreprises et leur intranet ou bien les particuliers sur Internet, devant la quantité considérable de documents disponibles, cela devient vite un parcours du combattant.

C'est donc pour pallier à ce manque que les grands éditeurs informatiques tels que Yahoo, Google ou Microsoft se sont lancés dans le développement de moteurs de recherche afin de pouvoir effectuer des requêtes et trouver les documents ou sites répondant aux mieux aux besoins de l'utilisateur. Malheureusement, ces moteurs de recherches, basés sur une recherche plein texte trouve ses limites maintenant devant la quantité de documents à traiter et certains « détournements » des métadonnées par des propriétaires de sites peu scrupuleux afin d'augmenter leur trafic et dégager plus de revenus ou enfin la richesse linguistique (un même mot peut avoir plusieurs significations totalement différentes).

Il sera étudié dans cette section les objectifs du web sémantique, des langages et outils utilisés afin de pouvoir remplir les objectifs fixés et enfin, les outils existant afin de visualiser ces systèmes complexes sémantiquement structuré.

### 2.1.2 Objectifs du Web Sémantique

C'est dans ce constat un peu alarmiste, comme nous venons de l'introduire, que le web sémantique se développe de plus en plus. L'idée étant de donner une sémantique à tous les documents afin de pouvoir ensuite utiliser des techniques plus poussées dont nous parlerons par la suite. Ainsi, lorsque l'on tape le mot "pêche" dans un moteur de recherche, on trouve des résultats se rapportant au fruit, au loisir, etc. . . Ainsi, avec le web sémantique il serait possible d'informer l'utilisateur qu'il existe plusieurs thématiques en rapport à sa recherche et donc la possibilité d'indiquer lequel l'intéresse.

Le web sémantique se base sur les métadonnées qui existent depuis longtemps sur le web (inventé en 1994 par Tim Berners-Lee au même moment que le W3C) mais la notion de web sémantique pour le grand public est assez récente. Le principe général du web sémantique est de se baser sur le web classique et donc de publier et consulter des documents mais le web sémantique traite des informations formalisées à la place de textes en langage naturel (français, anglais. . .) comme le web classique, dans le but d'auto-

matiser ces traitements [7].

Plus précisément, comme le dit Bénédicte Le Grand dans sa thèse [38], « L'objectif du Web Sémantique n'est pas de permettre aux machines de se comporter comme des êtres humains, mais simplement de développer des langages pour exprimer des informations d'une manière traitable par des machines. En effet, le traitement automatisé des données requiert une représentation de la sémantique compréhensible et échangeable par les machines. Le Web Sémantique apporte une structure aux informations contenues dans les pages Web, afin de permettre à des agents logiciels d'effectuer des tâches sophistiquées pour les utilisateurs. Par exemple, on souhaite rendre les moteurs de recherche capables, non seulement d'interpréter des requêtes en langage naturel comme "où puis-je commander un chéquier?", mais aussi de savoir de quelle banque l'internaute est client et de la lister en premier. L'un des objectifs du Web Sémantique est de permettre aux machines de rassembler des informations à partir de sources différentes et de "raisonner" à partir de ces données, plutôt que de simplement les afficher. Le Web Sémantique n'est pas un Web à part mais une extension du Web actuel, au moyen de structures sémantiques. Les premières étapes d'intégration du Web Sémantique dans la structure du Web existant sont déjà en cours, tout particulièrement dans le cadre du World Wide Web Consortium (W3C). »

Pour cela, le web sémantique se base sur des langages mis en place et normalisés par le W3C ou l'Organisation internationale de normalisation (ISO) que nous détaillerons par la suite.

Avant d'exposer les langages et outils développés pour le web sémantique, nous allons présenter l'architecture du web sémantique.

### 2.1.3 Architecture du Web Sémantique

Afin de comprendre les langages/outils qu'il a fallu développer, voici le schéma de l'architecture du Web Sémantique [38] et nous allons en expliciter les différents éléments. Comme pour les modèles du domaine des réseaux par exemple (modèle TCP/IP ou bien OSI) basés également sur des couches, nous avons le principe suivant : la couche N utilise les services N+1 et offre des services à la couche N-1 (en considérant que la numération des couches se fait de bas en haut).

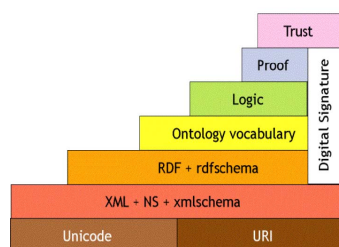


Figure 2.1 – Architecture du Web Sémantique

Dans la couche la plus basse du schéma, on retrouve le standard **Unicode et les URI** (*Uniform Resource Identifier*) qui sont utilisés pour coder les données brutes des pages web par exemple et de les identifier de manière unique. À ce niveau, aucune sémantique n'existe. Il est impossible de faire le moindre traitement automatique afin d'extraire des informations des documents ou bien d'effectuer des raisonnements (de type inférence par exemple comme on le pratique dans le domaine de l'intelligence artificielle).

La seconde couche, **XML + NS + xmlschema**, permet de structurer les données et ainsi s'intéresser au contenu de la page sans devoir s'interroger sur la manière dont on souhaite l'afficher sur la page web. NS, pour namespace, est comme indiqué dans [30], une notion permettant de désambigüiser des termes qui pourraient être homonymes sans cela. Un espace de nommage est matérialisé par un préfixe identifiant de manière unique la signification d'un terme. Au sein d'un même espace de nommage, il n'y a pas d'homonymes. C'est donc un premier niveau de sémantique qui est malgré tout très léger car il ne permet toujours pas de faire des associations entre plusieurs éléments. C'est pour cela que le W3C, sous l'impulsion de

Tim Berners-Lee, développe le standard **RDF** et le **rdfschema** que l'on retrouve à la couche supérieure que nous développerons dans la section 2.1.4. De son côté, l'ISO définit le standard **Topic Maps** permettant d'accomplir la même fonction que nous développerons à son tour dans la section 2.1.5 page 11.

Les syntaxes du langage *RDF* ou *Topic Maps* permettent d'affecter des propriétés à des objets, mais ces annotations sémantiques ne suffisent pas [38]. Il est par exemple impossible d'indiquer des transitivités dans les propriétés ou bien il n'est pas possible d'indiquer que deux concepts sont identiques mais avec des identifiants différents (souris et mouse) ou bien à l'inverse des identifiants semblables pour représenter deux concepts différents (pêche comme un fruit ou bien comme le loisir). C'est donc pour pallier ce manque que l'on a développé les **ontologies** qui se situe sur la 4<sup>ème</sup> couche de l'architecture. Nous reviendrons sur ce concept dans la section 2.1.6 page 12.

Étant donné le but du web sémantique : échanger et partager des données à partir de sources variées, de les traiter, même entre services qui ne sont pas forcément compatibles, il est parfois nécessaire d'avoir recours à un système d'authentification et de vérification de l'intégrité des informations partagées. On peut utiliser pour cela, une signature électronique qui apparaît sur plusieurs couches de l'architecture.

En ce qui concerne les 3 dernières couches (logic, proof, trust), elles ne seront pas étudiées dans ce mémoire car l'intitulé est suffisamment clair pour savoir en quoi consiste les services et ce n'est pas le sujet de ce travail. Nous allons maintenant étudier plus en détails les couches 3 et 4 consacrées au standard RDF et aux ontologies. La couche 2 consacrée à XML n'est pas détaillée car il s'agit d'un domaine suffisamment répandu dans l'informatique pour que chaque informaticien connaisse son principe. Pour plus de détails, on peut néanmoins se rendre sur la page du W3C consacré à XML [1].

## 2.1.4 RDF

[38] nous présente le langage RDF (*Resource Description Framework*) comme un langage qui a été développé dans le but de faciliter la création et l'échange de descriptions de ressources comportant de la sémantique sans en perdre le sens. Ce standard permet de traiter des métadonnées concernant des ressources de types très différents comme par exemple des documents PDF ou Word, des courriers électroniques, des fichiers audio et vidéo, des pages HTML ou des données XML. Une déclaration RDF est un triplet  $\langle \text{ressource}, \text{propriété}, \text{valeur} \rangle$  également aussi nommé  $\langle \text{sujet}, \text{prédicat}, \text{objet} \rangle$ . Chaque ressource est caractérisé par une URI et une ressource n'est pas obligatoirement un objet accessible sur le web (ce peut être par exemple un livre). On peut par exemple reprendre l'exemple des métadonnées de [11] qui est l'auteur, la date de création et de modification d'une page web. Si nous avons par exemple une page web appartenant à John Smith qui a été créée le 1<sup>er</sup> janvier 1999 et modifiée le 1<sup>er</sup> août 2004, la déclaration RDF de l'auteur de la page internet (ressource) est le triplet suivant :  $\langle \text{http://www.foo.com/~smith}, \text{author}, \text{John Smith} \rangle$ . Le triplet associé à la date de création est le suivant :  $\langle \text{http://www.foo.com/~smith}, \text{creation}, \text{January 1st 1999} \rangle$ . On peut représenter la représentation RDF de la page web de manière graphique par l'intermédiaire d'un graphe avec des nœuds et des arcs. Voici le graphe que l'on obtient pour notre page :

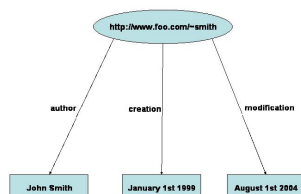


Figure 2.2 – Exemple d'un graphe RDF

La syntaxe du langage RDF repose sur XML et voici donc la traduction du graphe de la figure 2.2 page précédente en XML :

```
<?xml version=1.0"?>
<RDF>
<Description about="http://www.foo.com/~smith">
  <author>John Smith</author>
  <created>January 1, 1999</created>
  <modified>August 1, 2004</modified>
</Description>
</RDF>
```

Les déclarations RDF peuvent être également récursives. Ainsi, il est possible d'avoir une déclaration qui appelle une autre déclaration. Graphiquement, cela se représente sous la forme d'hypergraphe. Ainsi, nous pourrions avoir l'exemple suivant tiré de [38] :

```
<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <Creator rdf:resource="http://www.w3.org/staffId/85740"/>
  </rdf:Description>

  <rdf:Description about="http://www.w3.org/staffId/85740">
    <Email>lassila@w3.org</Email>
  </rdf:Description>
</rdf:RDF>
```

Ce code indique que la page *http://www.w3.org/Home/Lassila* (qui est le sujet) a été créée par l'employé n° 85740 (qui est l'objet). Dans la déclaration suivante, la même ressource (employé 85740) joue le rôle de sujet pour déclarer que son adresse email est *lassila@w3.org*. Graphiquement, nous obtenons la figure 2.3.

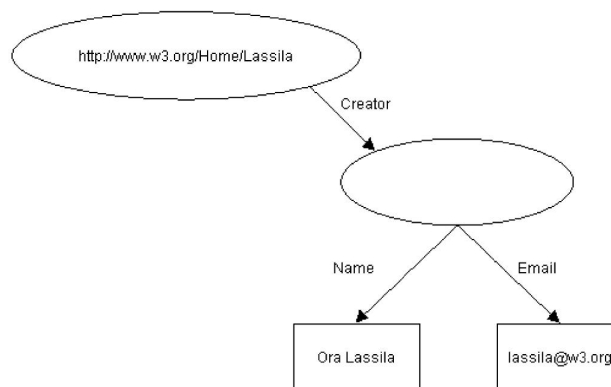


Figure 2.3 – Représentation graphique de la déclaration RDF



## 2.1.5 Topic Maps

Les Topic Maps sont un standard de l'ISO et non pas du W3C comme pour le RDF. Les objectifs sont assez similaires et selon [6], les cartes topiques (en anglais Topic Maps) sont un outil très général de représentation de connaissances, dont le but est d'agréger autour d'un point unique d'indexation (appelé topic) toutes les informations disponibles concernant un sujet donné, et de relier ces points par un réseau sémantique de relations appelées **associations**.

Le début des Topic Maps remonte aux années 1990 autour d'un groupe nommé Davenport. Ce groupe travaillait autour du domaine de l'édition et ils ont développé les Topic Maps pour permettre la construction d'index, de glossaires, de thésaurus et de tables des matières afin de permettre l'échange de documents informatisés. Ce groupe est également à l'origine de DocBook, l'une des DTDs les plus utilisées pour la création de documents SGML et XML.

Maintenant, l'utilisation des Topic Maps dépasse largement son champ d'application de départ. Afin de parvenir à utiliser les topic maps dans le domaine du web, un groupe de travail a été créé pour adapter le paradigme des Topic Maps reposant sur la syntaxe XML et on les appelle XML Topic Maps (XTM) [38]. Les Topic Maps reposent sur 3 concepts qui sont les **topics**, les **associations** et les **occurrences**. Malgré la simplicité de ces 3 éléments, il est possible de représenter des structures très compliquées et étendues. Voici un exemple de Topic Maps tiré de [38] :

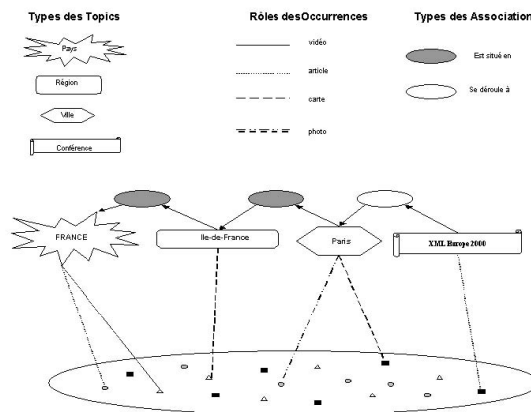


Figure 2.4 – Exemple de Topic Map

Dans une Topic Map, un topic est un objet qui représente un sujet. Dans le cas de l'exemple de la figure 2.4, un Topic pourrait représenter un sujet tel que le pays France, la région Île-de-France, la ville de Paris ou le congrès XML Europe'2000 qui s'est tenu à Paris. Dans une Topic Map, tout topic est une instance de zéro, un ou plusieurs types de topics ; c'est-à-dire les types représentent une relation de classe à sous-classe. Par exemple, dans le cas d'un thésaurus, les topics pourraient représenter des termes, des significations et des domaines, dans le domaine de la documentation information, il pourrait s'agir de fonctions, de variables, d'objets, de méthodes.

L'occurrence est un lien vers une ressource sur le sujet du Topic. Les occurrences sont classifiables par type : document texte, image, statistiques, ... L'avantage des Topic Maps, c'est qu'elle ne modifie pas les documents qu'elles indexent et il est donc possible de créer plusieurs Topic Maps pour un même ensemble de ressources.

Enfin, les associations de topics permettent de décrire des relations entre les concepts et donc cela permet de passer à un niveau au-dessus d'index simples. On peut ainsi par exemple représenter des relations qui apparaissent dans les thésaurus (*terme plus général, terme plus spécifique, terme relié*), exprimer les relations des tables de BD relationnelles, ... Sinon nous reprenons notre exemple de la figure 2.4, on

peut exprimer des associations comme *XML Europe'2000 se déroule à Paris, Paris est située en Île-de-France, l'Île-de-France est située en France, ...* Dans [11], il est précisé que chaque topic impliqué dans une association joue un rôle spécifique à cette association. Ainsi, dans l'association *Paris est située en Île-de-France*, le topic *Île-de-France* joue le rôle de contenant tandis que *Paris* joue le rôle de contenu. Il est également intéressant de noter que les topics et les ressources appartiennent à deux couches différentes. Il est ainsi possible de naviguer à un niveau abstrait (topic) au lieu de devoir naviguer directement dans les données.

Ainsi, pour conclure sur les Topic Maps et faire une analogie avec RDF, les Topic Maps sont plus orientées sur la navigation à l'intérieur de données et voir les relations qui existent entre les différentes ressources tandis que RDF est plus orienté vers l'interrogation et la rédaction de requêtes. Il existe en effet plutôt des langages d'interrogation plutôt que RDF même si les deux standards sont compatibles comme il a été démontré dans [43].

### 2.1.6 Ontologie

Lorsque l'on cherche à définir une ontologie, on se trouve confronter à plusieurs définitions liées à des domaines complètement différents :

- la philosophie où l'ontologie est une branche de la métaphysique
- en médecine où l'ontologie est l'intérêt pour la genèse des maladies
- et enfin en informatique où il s'agit d'un système de représentation des connaissances

Ici, le domaine qui nous intéresse est évidemment celui de l'informatique. Ainsi, dans [3], on trouve la définition suivante : « En informatique, une ontologie est un ensemble structuré de concepts. Les concepts sont organisés dans un graphe dont les relations peuvent être :

- des relations sémantiques ;
- des relations de composition et d'héritage (au sens objet)

L'objectif premier d'une ontologie est de modéliser un ensemble de connaissances dans un domaine donné. »

Du point de vue opérationnel, [3] définit l'ontologie comme « un réseau sémantique qui regroupe un ensemble de concepts décrivant complètement un domaine. Ces concepts sont liés les uns aux autres par des relations taxonomiques (hiérarchisation des concepts) d'une part, et sémantiques d'autre part. » Grâce à cette définition, il est possible d'écrire des langages destinés à implémenter des ontologies. On peut par exemple citer des langages comme N3 (N-Triple)<sup>1</sup> ou bien DEF-\* qui sont basés sur la **logique du premier ordre**.

De son côté, le W3C a travaillé sur le développement de langages de modélisation pour le web sémantique comme l'OWL<sup>2</sup> basé sur RDF précédemment développé à la section 2.1.4 page 9 et il a également spécifié une syntaxe XML afin d'en faciliter l'utilisation et l'échange. OWL s'inspire de langages précédents comme DAML+OIL, RDFS ou bien la logique de description. Pour plus d'information sur ce dernier point, vous pouvez consulter [5] ou bien [30].

Maintenant, du fait de sa facilité et de son caractère expressif, OWL est le standard le plus utilisé. Nous ne rentrerons pas plus dans les détails car ce ne sera pas un élément majeur de ce travail et nous ajouterons juste quelques éléments supplémentaires si besoin est par la suite.

Pour conclure le sujet des ontologies, quelques mots sur les composantes d'une ontologie, basés sur le mémoire de Thierry Gillain [30]. Bien qu'encore problématique, la question du contenu d'une ontologie partagent les experts mais néanmoins, ils arrivent à s'accorder sur le centre de l'ontologie dans lequel on retrouve :

- un ensemble de termes importants d'un domaine
- une spécification de la sémantique de ces termes au moyen d'une logique

De plus, il existe des mécanismes pour construire les futures ontologies qui sont les suivants :

- des classes
- des propriétés

<sup>1</sup><http://www.w3.org/DesignIssues/Notation3>

<sup>2</sup>Web Ontology Language

- des individus
- des fonctions
- des axiomes
- la composition

Si l'on souhaite plus d'informations sur le rôle de chacun de ces mécanismes, on peut se reporter au précédent mémoire cité [30] et les références qui y sont données également. En conclusion ainsi que pour illustrer et se donner une idée des relations existantes entre ces différents éléments, voici un exemple donné dans le travail de Thierry Gillain à la figure 2.5.

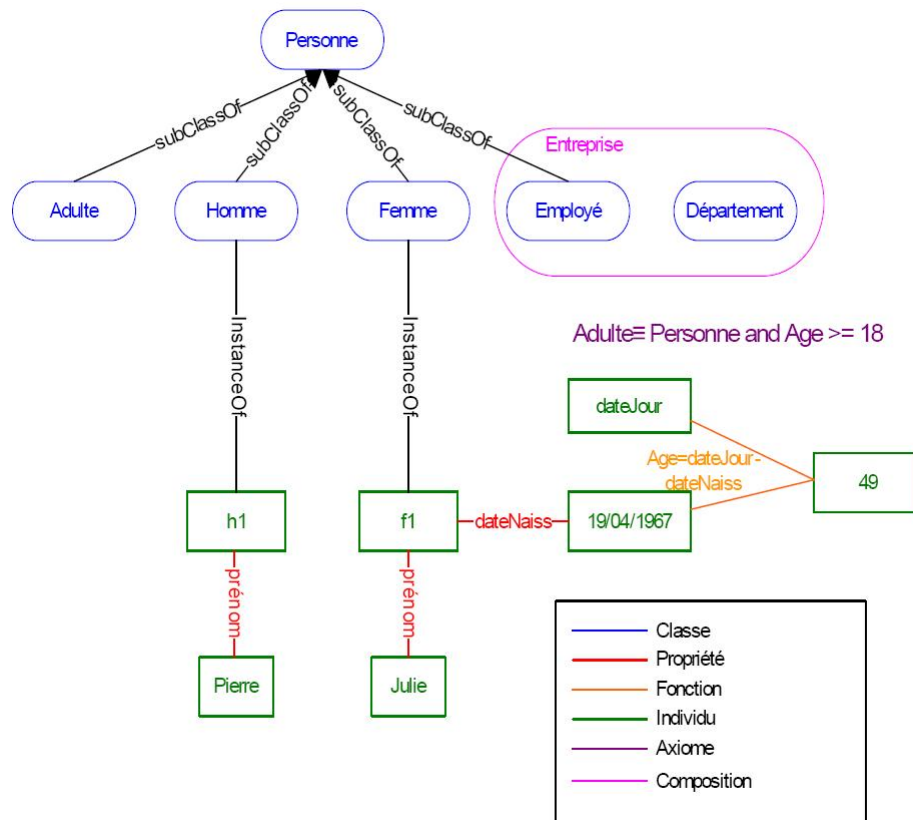


Figure 2.5 – Illustration des composantes traditionnelles d'une ontologie

### 2.1.7 Visualisation de systèmes complexes sémantiquement structuré

#### Ontopia

Ontopia<sup>3</sup> Omnigator est une application qui permet de parcourir des Topic Maps avec un simple navigateur internet.

Il se compose d'une application web Java J2EE installée sur un serveur Tomcat. Cette application a été développée à l'aide du Framework fournit dans l'Ontopia Knowledge Suite (OKS).

Sur Omnigator, l'exemple développé par Ontopia, ils ont laissé apparaître clairement les différents niveaux des Topic Maps (topic, association, rôle d'association, occurrence) afin de permettre au testeur de bien voir tout ce que l'on sait gérer grâce à l'application mais il est parfaitement possible de créer un site sans que l'utilisateur final se rende compte que derrière tout cela, se trouvent des Topic Maps comme il est possible de le voir dans les exemples de sites données plus loin.

**Test d'Ontopia** Lorsque l'on a choisi un Topic Map particulier sur lequel travailler (par exemple opera.xtm<sup>4</sup> ici en l'occurrence), on arrive sur la page "Ontology" qui répertorie alphabétiquement les topics, les associations, les rôles d'association et les occurrences qui existent dans le Topic Map.

Le "master index" affiche par ordre alphabétique tous les topics du Topic Map. L'"index of individuals" affiche tous les topics à l'exception des typing topics. L'"index of themes" affiche tous les topics utilisés comme thèmes dans le Topic Map afin de définir son scope (aussi parfois appelé scoping topics).

Les deux sous-menus de "Hierarchies" permettent d'afficher un tree view des associations conforme au "hierarchical relationship design pattern" de Techquila<sup>5</sup> et donc nous n'avons pas besoin d'abuser de liens supertype-sous-type afin d'afficher des relations hiérarchiques sous forme de tree view).

Ensuite, si on choisit un topic, on obtient une nouvelle fenêtre avec toutes les informations disponibles à son sujet (le résultat obtenu dépend du type du topic sélectionné). Par exemple si on choisit "Composer", on obtient la traduction dans différentes langues qui sont définies dans le scope. On obtient le(s) association(s) rattachée(s) au topic, sa hiérarchie. Nous avons également le "subject identifiers" qui permet d'avoir le ou les URI(s) pour le sujet sélectionné afin d'éviter les problèmes des homonymes ou synonymes, particulièrement lorsque l'on essaye de fusionner plusieurs Topic Maps ensemble. Enfin, nous avons également la liste des topics appartenant au type sélectionné.

Maintenant, si on sélectionne un topic individuel (en bas d'une hiérarchie), nous n'obtenons pas les mêmes informations à l'écran. On a par exemple la liste des types pour lequel le topic est une instance (ici, seulement Composer), nous avons la liste des occurrence internes (ici les dates de naissance et de mort par exemple) ainsi que la liste des occurrences externes (comme des liens vers des articles électroniques, des images, etc...). On retrouve comme pour le précédent topic, le "subject identifiers", le(s) association(s) rattachée(s) au topic.

**Vizigation** est un applet de Ontopia qui permet d'afficher graphiquement le Topic Map. On voit ainsi le topic sélectionné et tous les topics qui s'y rapportent directement. On a aussi une vue différente des informations comme on peut le voir à la figure 2.6 page suivante. On peut ensuite se déplacer dans le Topic Map en cliquant sur d'autres topics, on peut également rechercher un terme précis parmi les topics affichés, on peut choisir le nombre de niveaux à afficher à l'écran, etc... C'est donc le plus du logiciel et ce qui nous intéresse principalement dans notre cas car afficher le contenu d'un Topic Maps à l'écran, sous forme textuel n'est pas vraiment notre centre de recherche dans le cas présent. Ce qu'il faut donc retenir, c'est qu'on peut, grâce au framework de développement fournit dans Ontopia, développer un site web « classique » mais avec des Topic Maps derrière ou bien, grâce à l'applet Vizigation, avoir un affichage sous forme de graphe du contenu complet du Topic Maps. Néanmoins, vu qu'il s'agit de graphe, avec des arcs représentant les relations, les sommets les topics, etc..., il faut réserver cet affichage aux utilisateurs avancés ou bien aux personnes habituées à ce genre de représentation (mathématiciens, scientifiques, ...).

---

<sup>3</sup><http://www.ontopia.net>

<sup>4</sup>[www.ontopia.net/omnigator/models/topicmap\\_complete.jsp?tm=opera.ltm](http://www.ontopia.net/omnigator/models/topicmap_complete.jsp?tm=opera.ltm)

<sup>5</sup>[www.techquila.com/topicmaps.html](http://www.techquila.com/topicmaps.html)

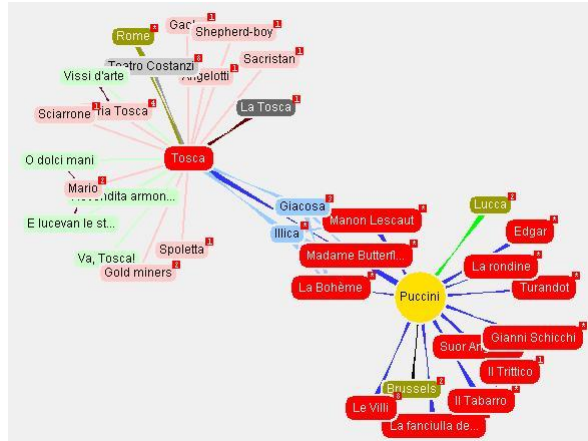


Figure 2.6 – Affichage d’un Topic Map avec Vizigation

Le langage utilisé par Ontopia pour les requêtes sur les Topic Maps est Tolog, un mélange de Prolog et de SQL. On peut trouver plus d’informations sur Tolog dans [4].

Voici deux sites utilisant Ontopia sans faire apparaître la moindre allusion à des topics maps qui sont derrière pour gérer le site :

- <http://www.kulturnett.no/>
- <http://www.ontopia.net/operamap/>

Il existe d’autres fonctions du côté concepteur à Ontopia comme des statistiques sur le Topic Map, la possibilité d’importer des Topic Map déjà créés, un outil pour créer son Topic Map, le modifier, ajouter des informations, etc... mais nous n’avons pas approfondi ce côté des choses car nous nous intéressons plus à la visualisation de l’information. Malheureusement, il n’est pas possible d’avoir accès gratuitement au Framework Java et voir un peu toutes les fonctions mises à notre disposition et la manière de les utiliser. On peut juste voir un résultat possible avec Omnigator ce qui est un peu dommage et difficile à juger mais sinon, il est vrai que Ontopia a l’air de proposer des outils intéressants et performants pour la manipulation de Topic Maps.

## TMNav

TMNav est un outil gratuit développé en Java. Il permet d’afficher des Topics Map que l’on importe au format XML. Ensuite, l’interface est composée de plusieurs blocs :

- à gauche : le ou les TM chargés et en dessous l’index de tout le contenu du TM
- à droite : un tree-view montrant l’arborescence du topic sélectionné et à côté, une vision graphique du TM et de ses différentes relations
- en-dessous, il y a une fente d’information

Dans le bloc d’index, on retrouve une liste déroulante permettant de sélectionner les éléments que l’on souhaite sélectionner : topics, associations, les types topics, les types associations, les types occurrence, les types membres. Une fois un élément sélectionné, en-dessous s’affiche les éléments correspondant et on peut ensuite sélectionner celui qui nous intéresse. Lorsque nous avons sélectionné un élément, les éléments correspondant s’affichent sous forme d’arborescence et on peut ensuite sélectionner celui qui nous intéresse afin de voir les différents topics en relation sous forme graphique.

Concernant l’affichage graphique, on peut choisir différents types de rendus : touchgraph par défaut mais sinon nous pouvons utiliser un rendu hypergraphe, un rendu section (visualisation sous forme textuelle) ou un rendu sous forme d’arbre.

Il existe également un moteur Tolog<sup>6</sup> pour effectuer des query précis dans le TM.

TMNav est un outil qui peut être intéressant si on l'utilise juste pour vérifier rapidement une information sur un TM et pour des utilisateurs avancés dans ce type de logiciel et qui manipulent aisément des TM. On ne pourrait malheureusement pas utiliser cet outil pour le grand public car même l'affichage graphique, n'est pas compréhensible pour une personne qui ne sait pas ce qu'est un TM comme par exemple dans le cas qui nous intéresse, un utilisateur qui souhaite planifier ses vacances.

Voici maintenant quelques captures d'écran afin d'avoir une idée des possibilités de ce logiciel qui est plutôt réservé aux chercheurs dans le domaine des Topics Maps plutôt qu'à des utilisateurs lambda à qui on voudrait afficher le contenu d'un Topic Map.

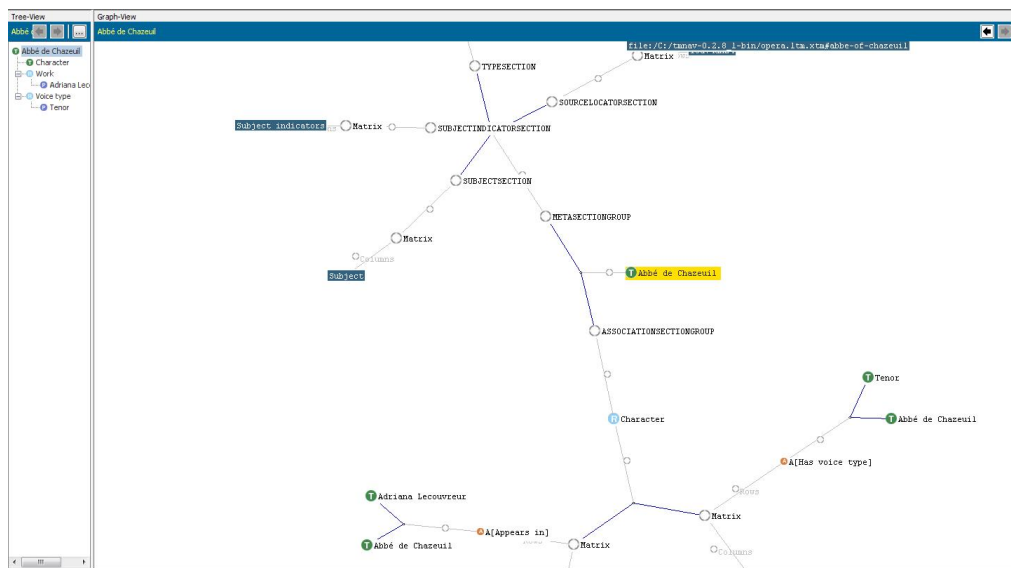


Figure 2.7 – Affichage d'un Topic Map avec TMNav dans le mode « Touchgraph »

La figure 2.7 montre l'affichage du Topic Map Opera fournit en exemple par Ontopia afin de pouvoir comparer les deux sur le même contenu. Ainsi, on voit au centre, le topic sélectionné (Abbé de Chazeuil) avec toutes les relations qui lui sont liées. Un inconvénient, en plus d'une difficulté à comprendre le graphe pour un utilisateur qui ne connaît pas la manière dont a été créé le Topic Map, c'est le chevauchement de certains éléments les uns sur les autres et l'impossibilité de déplacer les éléments sur l'écran.

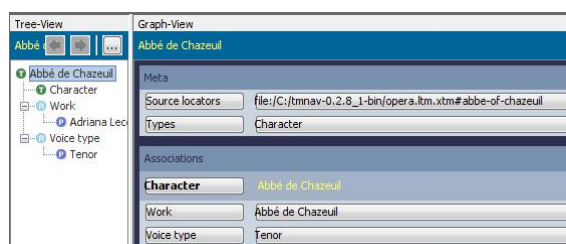


Figure 2.8 – Affichage d'un Topic Map avec TMNav dans le mode « Section »

La figure 2.8 montre l'affichage du Topic Map sous forme textuelle. On voit sur la gauche un tree view des différentes relations liées à l'élément sélectionné. On voit ici que l'élément sélectionné est de

<sup>6</sup>se reporter au paragraphe d'Ontopia

type character, joue un rôle Work avec Adriana Lecouvreur et possède la voix de type Tenor. Ensuite, on peut sélectionner dans le tree view de gauche les différents éléments et cela modifie en conséquence l’affichage et il est possible de voir toutes les instances du type sélectionné par exemple.



Figure 2.9 – Menu de TMNav

Enfin, la figure 2.9 montre l’affichage du contenu du Topic Map sous forme de liste afin de pouvoir sélectionner plus rapidement l’élément que l’on souhaite et ensuite visualiser graphiquement les éléments qui lui sont liés. On peut afficher les topics, les associations, les types de topic, les types d’associations, les types d’occurrences et les types membres.

### Autres pistes de recherches

Pour conclure cette section consacrée à la visualisation de système complexe sémantiquement structuré, nous allons aborder rapidement les recherches actuelles dans ce domaine et qui verront peut-être le jour prochainement sur l’internet de demain ou dans de nouvelles applications.

Actuellement, plusieurs travaux se basent sur la métaphore de la ville virtuelle pour afficher les résultats d’une recherche sur le web. Au lieu d’afficher les résultats comme Google, c’est-à-dire, une simple liste de site web avec le titre, l’adresse et un cours extrait de la page en question, des équipes de recherches tentent d’afficher cela en 3 dimensions sous la forme d’une ville. Dans cette métaphore, chaque bâtiment représente un site web et ils sont regroupés par quartier qui symbolise un cluster. Enfin, les différents éléments sont reliés entre eux par des rues qui symbolisent une certaine similarité et permet d’afficher quel(s) mot(s)-clé(s) relie deux éléments entre eux.

De plus, les caractéristiques de l’affichage des bâtiments apportent un lot d’informations supplémentaires à l’utilisateur. En effet, la hauteur du bâtiment indique la taille du site (plus ou moins de pages, documents, ...), la couleur du toit peut indiquer si l’utilisateur a déjà consulté le site web ou pas. Sur certains projets, comme celui de [48], sur les murs des bâtiments, on affiche certaines images de la page, ou bien un extrait du contenu de la page, ... afin d’aider l’utilisateur à trouver plus rapidement l’information qui l’intéresse.

On peut citer les recherches de Madame Le Grand [37], de Monsieur Bonnel [14], d'une équipe du MIT citée précédemment [48], etc... Nous reviendrons un peu plus en détail sur l'affichage de résultats de requêtes dans les moteurs de recherches à la section 3.2 page 24. Pour illustrer cela, voici quelques images afin de se donner une meilleure idée :

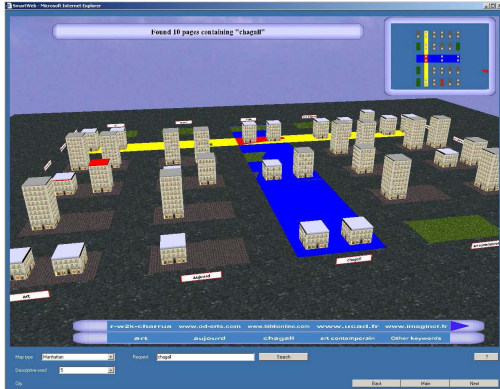


Figure 2.10 – Métaphore de la ville virtuelle [14]



Figure 2.11 – City of News [48]



### 3.1 Personnalisation/adaptation

Au début de l'informatique grand public, les éditeurs de logiciels ne se souciaient guère de l'avis des utilisateurs et les interfaces graphiques n'est pas leur soucis majeur. Depuis quelques années maintenant, les choses ont radicalement changé, l'arrivée de chercheurs dans le domaine des IHM, l'internationalisation des logiciels, etc. a conduit les éditeurs à écouter les utilisateurs qui sont par la même occasion leurs clients. Ainsi maintenant, la possibilité de personnaliser les interfaces et permettre aux utilisateurs de les adapter à leurs besoins et leur manière de travailler est un point essentiel des nouvelles applications (aussi bien les applications classiques que l'on installe sur son ordinateur que les applications web qui sont de plus en plus nombreuses). Nous allons donc nous intéresser, dans cette section à différentes manière de personnaliser une application. Dans un premier temps, on s'intéressera à la personnalisation d'interfaces ou d'applications, puis à la localisation de logiciels pour terminer par un exemple précis, la *nouvelle génération de moteur de recherche*.

#### 3.1.1 Personnalisation/adaptation d'interfaces/applications

Depuis maintenant plusieurs années, de plus en plus d'éditeurs de logiciels ou bien de sites web offrent à différents niveaux, un soupçon de personnalisation. Ainsi par exemple, de nombreux logiciels permettent maintenant de personnaliser les menus, les barres d'outils, etc. ... de la manière que l'on veut afin de pouvoir utiliser le logiciel comme on le souhaite vraiment. L'exemple le plus simple est par exemple avec la suite Office de Microsoft, dans les menus, les éléments que nous n'utilisons pas sont cachés automatiquement et il y a une petite flèche pour les faire réapparaître lorsque l'on en a besoin. Toujours dans la suite Office mais également dans de nombreux autres logiciels, il y a la possibilité de personnaliser les barres d'outils : c'est-à-dire déplacer les icônes, en supprimer, en rajouter, créer des groupes, des nouvelles barres d'outils, les mettre où l'on veut dans l'interface, etc. Cette tendance s'exporte maintenant sur le web surtout depuis l'apparition du Web 2.0 et des applications complètes disponibles en ligne.

Pour illustrer la personnalisation dans le domaine de l'Internet, nous pouvons nous intéresser à l'article [16] traitant de la personnalisation dans le domaine culturel et plus particulièrement celui des musées. Dans ce domaine, assez lié au domaine du tourisme qui nous intéresse, la personnalisation se situe plutôt au niveau des services offerts aux utilisateurs. Ainsi, en fonction de la connaissance de la personne dans le domaine traité sur la page et/ou le site web, on n'affichera pas les mêmes informations. Prenons l'exemple d'un tableau. Si la page affiche un tableau avec les informations sur ce dernier et que l'utilisateur est un

étudiant en art, on lui affichera un certain nombre d'informations techniques (comme le type de peinture, de toile, la méthode employée par le peintre, on pourrait aussi situer le courant artistique sur une frise, etc. ...) tandis que s'il s'agit d'un simple visiteur n'ayant pas de réelles connaissances artistiques, on affichera plutôt des informations sur la biographie du peintre, une description du tableau...

Sur le site du Metropolitan Museum of Art, la personnalisation est autre : chaque utilisateur a une sorte d'espace privée sur lequel il peut ajouter à un calendrier les expositions qui l'intéresse afin de recevoir un rappel au moment de l'exposition pour ne pas oublier d'aller la voir, il peut également créer ses propres galeries virtuelles en ajoutant les œuvres qu'il aime le plus pour les visualiser plus facilement quand il le souhaite, il peut également recevoir des newsletters sur les sujets qu'il aura choisi...

Enfin, la dernière personnalisation, la moins intéressante en terme de fonctionnalités utiles pour l'utilisateur, on trouve la "customization" qui consiste à modifier l'interface du site en fonction des goûts de l'utilisateur (police d'écriture, couleur, ...). Il s'agit donc simplement d'une personnalisation graphique de l'interface, pas au niveau des fonctionnalités offertes. On retrouve ce type de personnalisation de plus en plus fréquemment dans les applications bureautiques avec le système de "thèmes" graphiques qui permet de changer l'image des icônes, les couleurs de l'application, etc. ou bien sur Internet dans les forums de discussions, les blogs, etc. où l'on peut choisir différentes "skins" qui change les boutons, les couleurs, la police, la bannière, etc. sans ajouter de nouvelles fonctionnalités.

Une autre tendance enfin émerge également depuis quelques temps aussi : l'adaptation d'applications destinées aux ordinateurs de bureaux pour les périphériques mobiles comme les PDA. En effet, la mobilité est le maître-mot et de nombreux travailleurs sont plus souvent en déplacement qu'à leur bureau mais ils doivent néanmoins pouvoir accéder à leurs données n'importe où. Il existe bien les ordinateurs portables de plus en plus petits et puissants bien adaptés par exemples aux commerciaux mais pas pour des professions comme les médecins ou les infirmières lorsqu'ils sont au domicile du patient ou bien dans l'hôpital en faisant le tour des chambres. Ainsi, on commence à développer les applications, en plus du cadre classique des ordinateurs, pour les périphériques mobiles beaucoup plus petits. Deux alternatives s'offrent à nous pour cela : développer de zéro une nouvelle application, offrant les mêmes fonctionnalités que l'application existante et donc cela veut dire effectuer une analyse complète avec son propre cycle de vie et cela à chaque fois que l'on modifie trop l'application destinée aux ordinateurs classiques (ce qui implique beaucoup de travail, des coûts importants et une équipe d'informaticien à part entière) ou bien essayer de développer des outils qui permettraient de "convertir" l'application existante pour qu'elle puisse fonctionner correctement sur les périphériques mobiles. C'est dans la deuxième solution que des équipes de recherches se sont penchées afin d'essayer de développer un outil qui permettrait de générer le code de manière presque automatique à la fois pour l'application pour l'ordinateur classique et pour le PDA. On peut ainsi citer le projet SEFAGI [18] puis par la suite le projet SECAS [19]. Ainsi l'idée du projet a été de développer un framework basé sur des webservices permettant donc de générer automatiquement le code source des interfaces de l'application classique et pour les PDA. De plus, le système est capable automatiquement de connaître le type d'appareil utilisé pour consulter les données et donc d'adapter l'affichage en fonction de cela comme le montre la figure 3.1 page ci-contre et la figure 3.2 page suivante. La tâche principale maintenant est de définir les spécifications des interfaces que l'on souhaite (menu, boutons, photos, tableau, etc. ...) et ensuite, c'est le système qui se charge de générer le code des interfaces graphiques. Il ne restera donc plus qu'à écrire le code des fonctions du programme, sans se soucier des interfaces graphiques. C'est vers cette catégorie de développement qu'il est intéressant de se diriger et pourquoi pas essayer de développer un framework similaire pour des applications web.

Actuellement, les travaux d'adaptation pour les sites web sont plus orientés vers la manière de décomposer la page du site pour s'afficher d'une meilleure manière sur un PDA plutôt que de développer des outils pour générer automatiquement des pages web pour plusieurs contextes. Ainsi, dans la travail de recherche [42], l'équipe a travaillé sur la manière d'améliorer l'affichage des pages web sur les navigateurs web des périphériques mobiles ou tout du moins faciliter l'accès aux différentes parties de la page. La technologie « SmartView » s'intéresse à l'analyse du layout de la page et identifie pour l'utilisateur les différentes zones de la page qu'il a identifié. Pour cela, on affiche une miniature de la page à l'utilisateur et

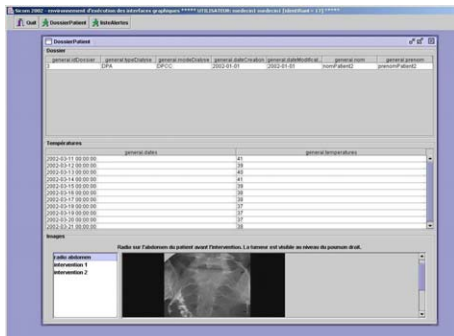


Figure 3.1 – Interface de la version classique [19]

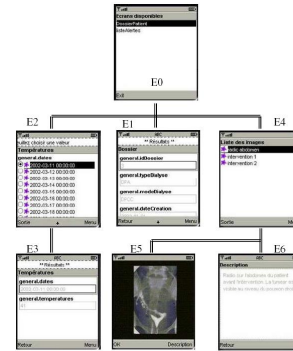


Figure 3.2 – Interfaces de la version mobile [19]

chaque zone identifiée est encadrée. Il suffit alors à l'utilisateur de cliquer dans la zone qui l'intéresse pour ne plus afficher que cette dernière en grand afin qu'elle puisse être lue par l'utilisateur. La technologie « SearchMobil » a pour but d'aider l'utilisateur à naviguer dans les résultats d'une requête sur un moteur de recherche depuis son périphérique mobile. En effet, les moteurs de recherches donnent un ranking à la page en fonction de l'adéquation de la page proposée par rapport à la requête de l'utilisateur. Le plus apporté par SearchMobil, c'est de fournir un ranking à l'utilisateur pour chaque partie de la page afin d'orienter plus rapidement l'utilisateur vers l'endroit où se trouve l'information qu'il recherche, sans qu'il doive lire l'intégralité de la page, ce qui n'est pas facile sur un périphérique mobile. Ainsi, en associant cette technologie à la précédente, on peut proposer à l'utilisateur l'affichage de la miniature de la page et pour chaque zone, indiquer le nombre de fois que les mots de la requête sont cités dans cette zone. Ainsi, chaque carré orange indique que le ou les termes sont cités une fois dans la zone. De plus, la zone avec le plus grand ranking est encadrée en rouge tandis que les autres zones le sont en vert. Ainsi, directement, l'utilisateur voit la zone de la page qui est statistiquement la plus intéressante pour lui. Enfin, SearchMobil offre une navigation simplifiée entre les 10 premiers résultats de la requête par un système d'onglet sur la droite de la fenêtre où il suffit à l'utilisateur de cliquer sur un des onglets pour afficher la miniature de la page avec les zones encadrées et le ranking pour chacune d'elle.

Une étude a été ensuite faite afin de vérifier la validité du système en le comparant par rapport à Pocket IE (navigateur web par défaut fournit sur les Pocket PC). Pour effectuer cette étude, on a donné à des candidats une série de questions auxquelles ils doivent trouver une réponse et on compare le temps que les candidats mettent pour trouver la réponse entre le système SearchMobil et Pocket IE. On a défini 3 types de questions :

1. la page peut être correctement divisée en zone par SearchMobil et la réponse à la question se trouve dans la zone encadrée en rouge
2. la page est découpée simplement en une seule zone
3. la page est découpée en plusieurs zones et la réponse ne se trouve pas dans la zone encadrée en rouge

Dans le premier cas, le temps pour trouver la réponse est meilleur pour SearchMobil par rapport à Pocket IE (10.59s de moyenne pour SearchMobil et 13.64s pour Pocket IE). Pour les deux derniers cas, le temps de réponse est légèrement supérieur pour SearchMobil que Pocket IE (14.86s de moyenne pour SearchMobil contre 14.11s pour Pocket IE dans le 2e cas et 19.52s contre 16.21s pour le dernier cas). La très faible différence de temps dans le 2e cas s'explique par le temps supplémentaire nécessaire pour décomposer en zone la page par SearchMobil alors que cela n'apporte rien puisque la page ne comportera qu'une seule zone. Pour le dernier cas, la différence est variable en fonction de la complexité de la page. Si la page est courte, le temps de réponse sera beaucoup plus favorable pour Pocket IE que SearchMobil tandis qu'avec des pages beaucoup plus longues, statistiquement, on peut attendre de meilleurs temps de réponses pour SearchMobil ou très proche de Pocket IE. Globalement, on peut conclure que le système développé autour de SmartView puis inclus dans le système SearchMobil, les résultats obtenus sont bons

et nous encouragent à poursuivre dans cette voie pour améliorer la visualisation de page web sur des périphériques mobiles.

### 3.1.2 Localisation

Même s'il s'agit d'un travail encore assez peu connu pour le grand public, la localisation de logiciels et/ou de sites web commencent à prendre de l'importance. Mais en quoi consiste réellement la localisation ? Ce qui vient en premier à l'esprit, c'est la traduction d'un logiciel ou d'un site web dans une autre langue. C'est vrai mais c'est loin d'être la seule tâche de la localisation et c'est un travail long et bien plus compliqué que ce qu'il ne paraît. En effet, en plus de la traduction, il s'agit d'adapter complètement le logiciel ou le site à la culture du pays pour lequel on fait le travail. Chaque culture a une manière bien à elle d'utiliser les ordinateurs, de voir les nouvelles technologies, etc. et donc il faudra en tenir compte durant la localisation. De plus, les symboles, les couleurs utilisées ont une signification bien particulière d'une culture à l'autre et donc ce que nous pourrions ressentir comme une couleur pour avertir d'une erreur pourrait être ressenti comme une offense dans la culture asiatique par exemple. Enfin, la traduction d'un logiciel en français en anglais par exemple n'est pas aussi simple qu'il n'y paraît. À première vue, on se dit qu'il suffit de traduire du français vers l'anglais tous les mots du logiciel. Mais c'est faux ! En effet, en fonction du pays pour lequel on veut la traduction en anglais, on ne traduira pas les mots de la même manière (différences entre l'anglais de Grande-Bretagne et l'anglais américain, différences dans les expressions utilisées, ...). Un autre problème se pose également : la longueur des champs que l'on a réservé dans l'interface graphique. En effet, la longueur d'un mot ou d'une expression n'est pas la même d'une langue à l'autre et donc il faudra bien veiller à ce que la traduction soit suffisamment courte pour tenir dans l'espace prévu (c'est souvent un problème dans la traduction de l'anglais vers le français qui est beaucoup plus expressif et donc avec des phrases et expressions plus longues).

Pour mieux se rendre compte des choses, nous nous baserons sur un travail de localisation pour le site web de FirstSearch. FirstSearch est un service en ligne pour les bibliothèques du monde entier. Au départ, l'interface de recherche et l'aide d'utilisation est en anglais mais un projet est lancé pour traduire l'interface et l'aide en français et en espagnol (pas le contenu de la base de donnée du site) afin d'éliminer la barrière de la langue pour l'utilisation du site. Lors de ce projet, l'équipe de First Search a été confrontée à plusieurs problèmes :

- l'architecture du système n'est pas optimisé pour la traduction
- en parallèle du projet de traduction, il y a un projet de mise à jour majeure du système
- la localisation (la traduction française est destinée pour la France, le Canada et les pays francophones tandis que la traduction espagnole est destinée à l'Espagne, aux pays d'Amérique Latine parlant l'espagnol et aux communautés hispaniques des États-Unis)
- ressources limitées pour le projet (à cause de l'autre projet en parallèle)

Ainsi, en s'intéressant de plus près au projet de FirstSearch, on s'aperçoit rapidement que ce n'est pas une tâche aisée que de se lancer dans la traduction d'une application de tout genre (même si on laisse tomber le problème du projet en parallèle qui a augmenté les difficultés). Ainsi, ce que l'on peut dire, c'est que quand on se lance dans le développement d'une application (web ou classique), il faut penser à la possibilité (même si ce n'est que dans un futur plus ou moins éloigné) de la proposer dans plusieurs langues. Il est donc nécessaire de « stocker » toutes les chaînes de caractères en dehors du code de l'application afin de pouvoir effectuer le travail en parallèle sans gêner le travail des deux équipes ou bien avoir la possibilité de faire sous-traiter la tâche à une entreprise spécialisée dans la traduction. Il est donc vivement conseillé de coder le texte des applications (tout ce qui est extérieur au code de programmation) dans des fichiers séparées. De plus, il existe des applications pour faciliter la tâche de localisation, lorsque nous avons bien pris le soin, en moment de la conception de l'architecture de la future application, de prévoir cela. On peut par exemple citer le logiciel TsiLang<sup>1</sup> qui offre des outils pour faciliter la tâche et il permet même de laisser la tâche aux utilisateurs, pour traduire le logiciel. Ce genre d'outil permet d'accélérer le processus de traduction mais également d'augmenter en qualité car un

---

<sup>1</sup>[www.tsilang.com](http://www.tsilang.com)

dictionnaire accompagne le logiciel pour stocker les traductions déjà effectuées et contrôler la cohérence des traductions (toujours employer la même traduction pour un mot donné dans un contexte donné). En plus, l'outil permet de contrôler la longueur de la traduction car on voit en permanence la taille occupée par la phrase initiale et la taille de la traduction. On peut ainsi en permanence contrôler que la traduction ne sera pas trop longue pour l'interface. En effet, il est courant de voir (même dans des logiciels de renommée internationale) que la traduction est trop longue et donc une partie est tronquée et ne s'affiche pas correctement.

Dans le cas du projet de FirstSearch, nous avons plutôt abordé les difficultés techniques liées à la localisation d'application. Pour conclure cette section, nous allons rapidement nous intéresser, par l'intermédiaire du travail de Kenneth Keniston [35], à l'aspect culturel qui est en jeu lors de la localisation d'applications.

Dans son essai [35], Kenneth Keniston définit la localisation ainsi : « Un logiciel localisé culturellement doit être un logiciel dont on ne peut pas faire la distinction d'un logiciel développé par un membre de cette culture ». On voit donc clairement que dans ce processus, le gros du travail n'est pas de traduire un mot par un autre mais de s'adapter à la culture pour laquelle on cible le produit. Ainsi, on ne devrait pas adapter un logiciel pour une langue donnée (par exemple de l'anglais vers le français) mais pour un pays voir même pour un groupe de personnes d'un pays ou d'une culture (par exemple adapté un logiciel américain pour la France, la Grande-Bretagne, le Japon, etc.). Les États-Unis dominent les trois-quarts du marché des logiciels mondial et donc à l'origine, les logiciels sont développés pour la culture américaine mais les sociétés de développement comprennent l'importance de l'enjeu de la localisation comme par exemple Microsoft, Apple, IBM, etc. mais uniquement dans l'aspect technique plutôt que culturel de la tâche. Néanmoins, localement, les institutions d'un pays comme l'Inde doivent-elles adopter des logiciels en bengali, en hindi ou en anglais ?

Si nous voyons un logiciel comme quelque chose qui n'est rien d'autre qu'un outil culturellement neutre pour résoudre des problèmes universels, la localisation n'est rien d'autre qu'un problème technique néanmoins complexe et coûteux. Malheureusement, la réalité est tout autre : la localisation n'est pas uniquement cela et nous avons du mal de séparer les difficultés techniques, culturelles et politiques comme en témoigne l'épisode de Microsoft avec la Chine en 1992 raconté par Keniston [35]. En 1992, Microsoft a ouvert un bureau à Pékin et s'est lancé dans la vente de ses logiciels en chinois. Microsoft a utilisé les caractères mandarins. Ces caractères étaient utilisés avant la pré-révolution chinoise et encore à Taiwan. Mais depuis 1949, le gouvernement chinois a décidé de modifier les caractères en introduisant les caractères simplifiés. Pour cette raison, les autorités chinoises ont été offensées par ce choix de la part de Microsoft. Pour résoudre ce conflit, Bill Gates a passé du temps en Chine et a lancé des projets de plusieurs millions de dollars dans ses centres de développement chinois en collaboration avec des universités pour développer ses logiciels (en particulier son système d'exploitation) mais aussi dans les domaines de la télévision interactive, la reconnaissance vocale et d'écriture, etc.

Le problème concernant Internet est encore plus vaste et compliqué. En effet, nous n'avons plus le cas de l'anglais vers une autre langue mais un ensemble de plusieurs langues vers les autres langues. Si nous prenons les 8 langues internationales majeures (chinois, anglais, espagnol, français, russe, arabe, hindi et swahili), nous avons 64 (8x8) localisations linguistiques et culturelles à développer contre 8 dans le cas d'un logiciel. À l'époque où Keniston a rédigé son article, il minimisait le problème car il disait que la majorité des utilisateurs du net pratiquait l'anglais suffisamment pour s'y retrouver. De nos jours, avec l'accès à Internet à de plus en plus de personnes, de nouveaux pays, de nouveaux publics (les retraités sont de plus en plus nombreux à s'initier à Internet), cette affirmation est moins vraie et donc le problème de la localisation du web devient plus important (surtout pour les sites commerciaux ou des grandes entreprises internationales) car nous sommes encore loin d'avoir des outils de traduction valables.

Pour plus d'informations concernant le sujet, il est possible de se reporter aux articles [36] et [20] ainsi que les différentes références que l'on trouve à l'intérieur.

## 3.2 Nouvelle génération de moteur de recherche

Comme nous l'avons introduit dans la section 2.1 page 7 consacrée au web sémantique, les moteurs de recherches actuels tels que Google, MSN ou Yahoo commencent à arriver aux limites de leurs possibilités face à la croissance exponentielle du nombre de pages disponibles sur Internet. Lorsque nous effectuons une recherche, la plupart du temps, nous obtenons des milliers de réponses qui ne correspondent pas souvent à ce que nous recherchons ou bien il faut parcourir tellement de pages de résultats que l'on finit par abandonner avant d'avoir trouvé notre réponse.

Dans ce domaine, le web sémantique pourrait être d'un grand secours afin d'améliorer la qualité des réponses fournies aux utilisateurs tout en améliorant également la manière d'afficher les résultats afin de remplacer les pages "austères" de résultats actuels. Dans cette optique, plusieurs sociétés se sont lancées dans des projets afin d'améliorer les moteurs de recherches actuels tout du moins dans la manière de restituer les résultats à la requête. Si l'on étudie une page de résultat fournie par Google, on s'aperçoit que les résultats proposés sont sans lien sémantique entre eux, il n'y a pas de classement thématique d'effectuer et l'affichage des résultats n'est que partiel (on affiche 10 résultats par page et il est donc impossible d'avoir une vue d'ensemble des résultats proposés). [14] propose donc dans sa thèse, d'effectuer un *clustering à la volée* des résultats afin de pouvoir cibler l'attente des résultats que l'on souhaite. Sur la figure 3.3, nous avons indiqué les différentes thématiques que l'on retrouve dans les résultats pour la recherche sur « orange ».

The image shows a screenshot of Google search results for the query "orange". To the right of the search results, there is a vertical legend with colored squares and text labels indicating different semantic categories. The results are grouped by these categories, with red bars separating the groups. The categories and their corresponding colors are: **opérateur** (red), **musique** (blue), **ville / tourisme** (green), and **fruit / cuisine** (purple). The search results include links to various websites such as Orange (operator), Orange World (music), Orange World (city/tourism), Orange World (fruit/cuisine), and Orange World (fruit/cuisine).

Orange - [ Translate this page ]  
Orange (ex-Wanadoo) propose des offres Internet haut débit (ADSL, ADSL2+, READSL) et bas débit. Filiale du groupe France Télécom.  
www.orange.fr - 69k - 19 Sep 2006 - Cached - Similar pages  
Météo - Prévisions Météo France - [ Translate this page ]  
louis - r orange.fr/vInfos  
accueil - www.orange.fr/Une/  
téléphone portable et services - [ Translate this page ]  
More results from www.orange.fr »  
(...)  
Téléphone portable et services mobiles ORANGE - [ Translate this page ]  
Achetez votre téléphone portable sur le portail Orange et découvrez les services mobiles sur votre téléphone.  
mobile.orange.fr - 3k - Cached - Similar pages  
Portail Orange - [ Translate this page ]  
Vous éprouvez des difficultés pour vous connecter à Orange World ? ... Orange vous propose de configurer votre téléphone à distance, en suivant nos ...  
mobile.orange.fr/0visteur/PV?PS=CHMOBCONFIG - 13k - Cached - Similar pages  
(...)  
Orange - Ville d'Orange en Provence. Site officiel de la mairie. - [ Translate this page ]  
Le site officiel de la Mairie de la ville d'Orange en Provence - France.  
www.ville-orange.fr - 11k - Cached - Similar pages  
choregies d'orange 2006 - [ Translate this page ]  
Festival de musique lyrique dans le théâtre antique d'Orange.  
www.choregies.asso.fr - 4k - Cached - Similar pages  
Orange Caraïbe - [ Translate this page ]  
Tarifs, couvertures et actualité de l'offre de téléphonie portable d'Orange aux Antilles Françaises et en Guyane.  
www.orangecaraibe.com/ - 43k - Cached - Similar pages  
orange.com - [ Translate this page ]  
Wanadoo est désormais Orange. Visitez le site Internet de votre pays pour en ... Orange, Equant et France Télécom se sont regroupés pour former Orange ...  
www.orange.com/francais/default.asp - 9k - Cached - Similar pages  
Orange Accueil Tourisme - Provence Web - [ Translate this page ]  
Ville d'art et d'histoire, Orange brille par son patrimoine historique exceptionnel. Dès l'entrée de la ville, vous vous retrouverez au coeur de la ...  
www.provenceweb.fr/aucluse/orange/orange.htm - 10k - Cached - Similar pages  
(...)  
ORANGE - office de tourisme - [ Translate this page ]  
ORANGE toutes les informations touristiques de l'office de tourisme pour préparer son voyage, son séjour à l'hôtel, ses vacances.  
www.tourisme.fr/office-de-tourisme/orange.htm - 11k - Cached - Similar pages  
Projet Orange - [ Translate this page ]  
Projet Orange, groupe, musique.  
projetorange.com/ - 8k - Cached - Similar pages  
Orange - Wikipédia - [ Translate this page ]  
Cette ville était la capitale de la principauté d'Orange. Principauté qui donna son nom à la Marson d'Orange-Nassau, actuelle famille royale néerlandaise ...  
fr.wikipedia.org/wiki/Orange - 16k - Cached - Similar pages  
Orange - [ Translate this page ]  
Bienvenue sur la boutique Internet. internet haut débit. téléphonie. télévision. Connectez votre maison. à vos envies. Vous êtes déjà client ...  
abonnez-vous orange.fr - 10k - Cached - Similar pages  
(...)  
ORANGE : tout sur l'orange, recettes, saison de l'orange sur L'... - [ Translate this page ]  
ORANGE : tout sur l'orange, recettes, saison de l'orange sur L'Internaute.  
www.internaute.com/femmes/cuisine/encyclopedie/fiche\_composant/145/orange.shtml - 64k - 19 Sep 2006 - Cached - Similar pages

opérateur  
musique  
ville / tourisme  
fruit / cuisine

Figure 3.3 – Un extrait de la liste de résultats proposée par le moteur de recherche Google en réponse à la requête « orange » (recherche réalisée sur le Web francophone) [14]

C'est dans cette voie que travaille la société Vivisimo<sup>2</sup> et son moteur de recherche Clusty<sup>3</sup>. Comme on peut le voir sur la figure 3.4 pour la recherche sur « orange », sur la gauche de la page, nous voyons la liste des topics associés à notre requête et il nous est donc possible d'affiner notre recherche en choisissant le cluster qui nous intéresse.

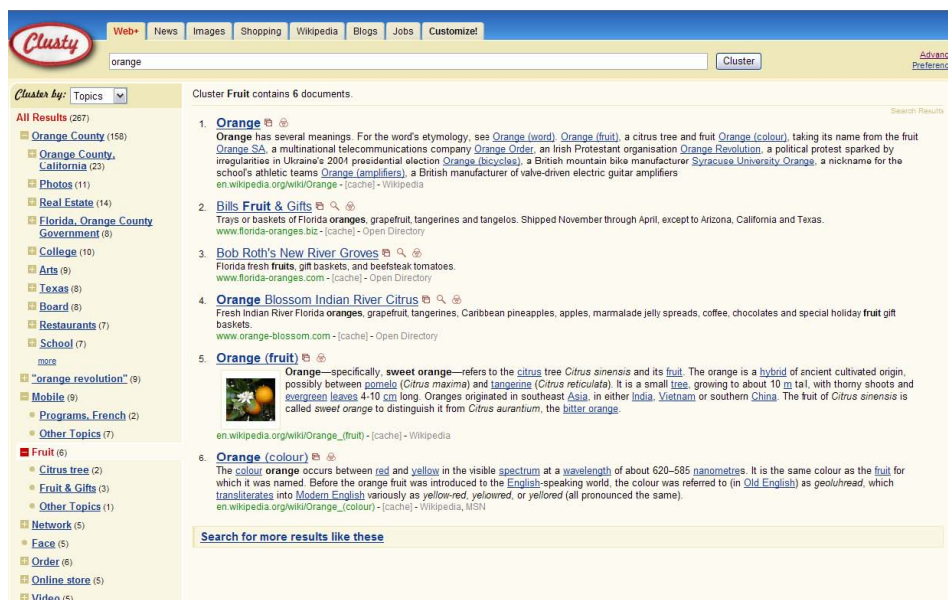


Figure 3.4 – Listes de résultats proposées par le moteur de recherche *Clusty* en réponse à la requête « orange » [14]

Néanmoins, le clustering effectué par *Clusty* est basé sur la similarité textuelle et linguistique et il n'y a aucun classement ensuite des réponses d'un cluster en fonction de proximité sémantique. Les résultats au sein d'un cluster sont affichés de la même manière que Google ou les autres moteurs de recherches du même type.

Pour pallier à ce manque, il existe une autre catégorie de travaux basé sur une approche cartographique basée sur la notion de graphes. Ainsi, on peut citer les moteurs de recherches *KartOO*<sup>4</sup> ou bien *MapStan Search*<sup>5</sup>. *KartOO* ne possède pas son propre moteur de recherche. En effet, il envoie la requête aux principaux moteurs de recherches et récupère la liste des résultats. Son travail commence alors en retravaillant la liste des résultats pour en dessiner une carte avec une description de la page et les mots-clés qui la concerne comme on peut le voir sur la figure 3.5 page suivante. Ainsi, on peut affiner sa recherche en ajoutant les mots-clés proposés par le site à notre recherche initiale et affiner notre requête au fur et à mesure. L'inconvénient du site, c'est qu'il est parfois nécessaire de dessiner plusieurs cartes avant d'avoir le sujet qui nous intéresse réellement.

En ce qui concerne *MapStan Search*, tout comme *KartOO*, il ne possède pas de moteur de recherche propre mais il utilise Google. Il retravaille ensuite les résultats reçus et archive chaque requête et résultats obtenus afin de pouvoir afficher ensuite la carte. Ainsi, il lui est possible de savoir quelle requête a permis de retourner plusieurs pages en même temps. Chaque site est représenté par un cercle plus ou moins grand en fonction de la taille du site et de couleur différente pour symboliser la pertinence par rapport à la requête (rouge pour les meilleurs et orange pour les intermédiaires et jaune pour les moins bons). De plus, l'affichage des résultats se basent sur la métaphore de la carte d'une ville en 2D où chaque place

<sup>2</sup>[www.vivisimo.com](http://www.vivisimo.com)

<sup>3</sup>[www.clusty.com](http://www.clusty.com)

<sup>4</sup><http://www.kartoo.com>

<sup>5</sup><http://search.social-computing.com/>



Figure 3.5 – KartOO, méta-moteur de recherche cartographique en 2D [14]

(rond symbolisant les sites) sont reliés entre eux par des rues afin d'indiquer un lien entre eux. Si on place le curseur de la souris sur une rue, la requête reliant les deux pages s'affichent (grâce au système d'archivage des requêtes et résultats indiqué au début) et plus la rue est large et plus de requêtes sont partagées par les sites. De plus, la longueur des rues indiquent la proximité des places. Plus la rue est longue et plus les résultats sont éloignés et retournés moins souvent en même temps par les requêtes et lorsque les deux places sont trop proches, elle fusionnent pour donner une place plus grosse. On peut voir la carte générée à la figure 3.6 page ci-contre pour la même requête que l'on avait effectué sur KartOO précédemment. Par la suite, il est possible de cliquer sur une place pour effectuer un zoom dessus et voir le contenu du site et les pages renvoyées en résultats. Ainsi, même si MapStan Search ne possède pas son propre moteur de recherche, il s'appuie sur sa propre base de connaissance qui s'enrichit au fur et à mesure des nouvelles requêtes pour dessiner sa carte des résultats. L'inconvénient du système, c'est la lenteur (parfois jusqu'à une vingtaine de secondes) pour générer les cartes de résultats. Néanmoins, ce système pourrait être utilisé dans d'autres domaines pour afficher des connaissances.



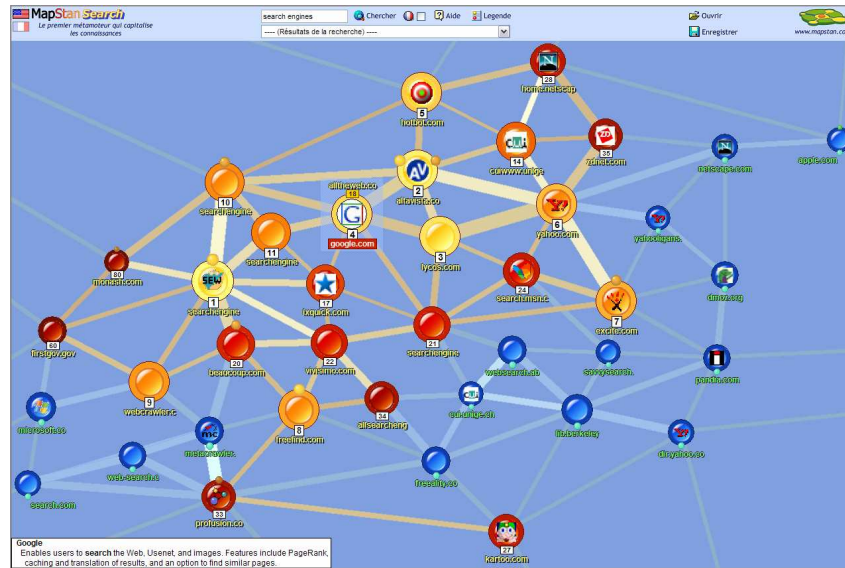


Figure 3.6 – MapStan Search, méta-moteur de recherche cartographique en 2D [14]

### 3.3 Caractéristiques et contraintes de développement du monde des périphériques mobiles

Le développement pour périphériques mobiles n'est guère différent que pour des applications bureautiques classiques. On dispose d'outils d'aide à la programmation, différents langages de programmation (comme le Java, C#, etc.). La seule différence à prendre en compte se situe du côté technique. En effet, les performances des périphériques mobiles sont beaucoup plus faibles que les ordinateurs de bureau (surtout au niveau du processeur et de la vidéo) et l'écran est beaucoup plus petit en taille et en résolution. De plus, les périphériques mobiles sont la plupart dépourvus de clavier (il existe un clavier virtuel que l'on utilise avec le stylet) et donc c'est surtout l'ergonomie et l'IHM qui est à revoir et à étudier de prêt plutôt qu'à la manière de programmer.

Pour commencer, une première difficulté que l'on rencontre lorsque l'on se lance dans le développement sur périphériques mobiles en Java, c'est la machine virtuelle. En effet, jusqu'il y a peu, un problème concernant le Java existait car bien que Sun a développé une version « allégée » de son framework Java (Java Micro Edition ou JME) mais il ne fournit pas, contrairement à la version destinée aux ordinateurs de bureau, de machine virtuelle pour exécuter du code Java sur ces périphériques mobiles. Il existe bien quelques outils payants, non compatibles sur tous les modèles de PDA ou bien des alternatives comme ewe<sup>6</sup> ou bien superwaba<sup>7</sup> qui permettent de développer des applications java sur des périphériques mobiles mais dans le cas du premier, le système n'est plus mis à jour depuis 2005 et pour le deuxième, une version est gratuite mais ne permet pas de faire tout ce que l'on veut. La version professionnelle du système coûte 495\$ par an. Depuis peu, un projet nommé *Mysaiifu JVM*<sup>8</sup> commence à devenir mature et la dernière version, la 0.3.3 date de décembre 2006. Ce projet libre fournit une machine virtuelle Java pour les Pocket PC<sup>9</sup> fonctionnant sous Windows Mobile 2003 et les versions suivantes. En plus, cette machine virtuelle permet de faire fonctionner les programmes Java écrit avec la version standard (J2SE) du framework de Sun qui est celui utilisée par défaut pour les applications bureautiques. Nous ne sommes

<sup>6</sup><http://www.ewesoft.com/>

<sup>7</sup><http://www.superwaba.com.br/>

<sup>8</sup>[http://www2s.biglobe.ne.jp/~dat/java/project/jvm/index\\_en.html](http://www2s.biglobe.ne.jp/~dat/java/project/jvm/index_en.html)

<sup>9</sup>PDA fournit avec le système d'exploitation de Microsoft, Windows Mobile

donc pas limités dans les fonctionnalités offertes par la version Micro. Cette contrainte ne nous bloque pas puisqu'il n'est pas prévu de devoir développer des applications mais uniquement un système web (utilisable depuis un navigateur internet).

La deuxième difficulté concernant directement les applications et sites web (ce qui nous intéresse dans notre travail), c'est l'affichage. En effet, la taille des écrans est relativement petite (entre 3.5 et 3.7 pouces, peut-être 4 pouces mais c'est vraiment le maximum) et de même pour la résolution (la résolution la plus courante est de 240x320 px et les modèles de plus haut de gamme en VGA est de 480x640 px mais avec un écran de même taille) pour les PDA. Pour les téléphones portables évolués (Smartphone), l'écran et la résolution sont encore plus petits.

Ensuite, la troisième contrainte concerne les performances de ces appareils. Même si pour des appareils de cette taille, les performances sont plus qu'acceptables, il ne faut pas espérer retrouver la même puissance qu'un ordinateur de bureau ou bien un ordinateur portable. Il est donc difficile d'envisager, par exemple, un affichage 3D comme par exemple dans la symbolique de la ville virtuelle car les processeurs ne sont pas assez rapides et puis la carte vidéo n'est pas suffisamment évoluée pour gérer l'affichage 3D à part un ou deux modèles de PDA haut de gamme à plus de 400€ (comme l'Axim x51v ou l'iPaq hx5400 ou hx2790 maintenant que le premier modèle n'est plus disponible).

Une autre contrainte à prendre en compte et qu'il ne faut pas négliger concerne les plugins ou autres extensions que l'on a l'habitude d'utiliser pour les sites web classique (Flash, musique, shockwave, etc...) qui n'existent pas forcément sur périphériques mobiles ou bien alors en version plus ancienne ou allégée. Par exemple, actuellement, le plugin Flash existant pour Pocket PC est en version 6 (7 depuis quelques jours) tandis que pour les ordinateurs classiques, nous en sommes à la version 9. Il faudra donc tester le fonctionnement des animations Flash, si on décide d'y avoir recours pour le site adapté aux périphériques mobiles.

Enfin le dernier point qui est le plus contraignant, c'est le navigateur web fournit par Microsoft sur son système d'exploitation pour périphériques mobiles (Pocket Internet Explorer) et les standards qu'il supporte. Après des recherches sur Internet, il semblerait qu'il soit équivalent à Internet Explorer 5 au niveau des standards et protocoles supportés. Nous verrons cela plus en détail au moment de faire mes tests d'affichages des résultats. A l'heure actuelle, il n'existe pas vraiment d'autres concurrents au navigateur de Microsoft. Il existe le navigateur Opera Mobile, développé par la société norvégienne qui édite déjà le navigateur pour les ordinateurs classiques, mais la version mobile est payante (19 €) ce qui doit sûrement dissuader les utilisateurs. Néanmoins, pour les utilisateurs assidus du web sur PDA, c'est l'alternative la plus intéressante car ce navigateur gère les standards les plus récents du web et permet donc de développer des sites beaucoup plus perfectionnés ou d'afficher les sites web non adaptée pour les périphériques mobiles d'une meilleure façon.

La fondation Mozilla développe de son côté une version mobile de son navigateur Firefox, dans le projet portant le nom « Minimo »<sup>10</sup>. Au début de la rédaction du mémoire, le projet n'avait pas l'air d'avancer très fort mais en allant voir aujourd'hui pour l'adresse du projet, nous avons remarqué qu'il avait fait un site assez succinct qui n'existait pas auparavant pour le logiciel. Nous sommes encore loin visiblement d'une première version finale (la dernière version est la 0.016 datant du 12 juin 2006 mais depuis, des versions beta plus récente, dite « nightly builds », existent). Mais tout récemment, la version 0.2 est sortie (on peut la considérer comme une version finale qui offre les fonctionnalités suivantes : compatible avec le Javascript, la technologie AJAX « web 2.0 », la gestion améliorée des favoris, la navigation par onglet, le support du RSS, le tout en plus du support des extensions et des widgets de Firefox).

Ceci permettra peut-être enfin de faire évoluer les possibilités du web sur les périphériques mobiles et un meilleur affichage des pages grâce au travail d'Opera et de la fondation Mozilla.

---

<sup>10</sup><http://www.mozilla.org/projects/minimo/>



Figure 3.7 – Exemple sous Opera 8.65 Beta

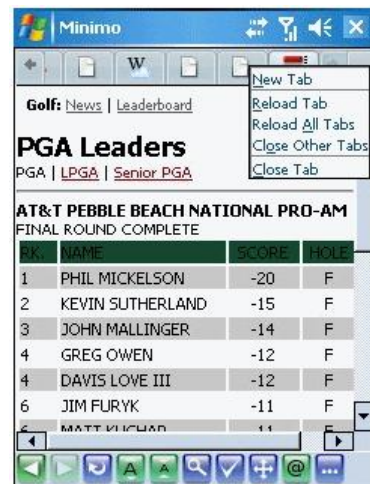


Figure 3.8 – Exemple sous Minimo 0.2



---

### Présentation du domaine touristique et de l'existant

---

Le domaine touristique est un domaine vaste et qui existe depuis longtemps. À ses débuts, son support principal était le papier mais depuis la croissance importante d'Internet, le tourisme s'y développe de plus en plus. De nos jours, de nombreuses villes possèdent leur propre site afin de faire découvrir aux touristes potentiels son histoire et son patrimoine. Nous trouvons également de nombreuses agences de voyages et même les célèbres guides touristiques (Michelin, guide du routard, etc.) passent à l'air du numérique, en compléments des guides papiers existant depuis des dizaines d'années.

Afin d'avoir une idée de tout ce qui existe actuellement dans le domaine du tourisme (aussi bien sur des supports traditionnels et que sur le web), nous allons faire un « état de l'art » en plus de l'analyse du site de la Nièvre qui nous sert un peu de support dans nos travaux de recherches (il s'agit d'une plate-forme pilote du portail touristique que l'on souhaite développer et adapter aux petits écrans des périphériques mobiles).

Nous terminerons ce chapitre par un bref aperçu des nouveaux supports qui se développent de plus en plus de notre quotidien et que nous pourrions utiliser pour naviguer sur notre futur portail.

#### 4.1 Analyse du site de la Nièvre

Le partenaire privé du projet Eiffel, Mondeca, a déjà développé un projet « pilote » avec la Nièvre pour développer leur site web<sup>1</sup> basé sur une ontologie du tourisme pour orienter les recherches de l'utilisateur. Cet exemple permet d'avoir une première idée sur la manière d'intégrer des services sémantiques dans un site grand public.

L'idée étant d'adapter le site pour les périphériques mobiles, il est intéressant d'essayer de naviguer sur le site depuis un PDA. En l'absence d'un tel périphérique, il a été nécessaire d'installer et configurer l'émulateur de Pocket PC développé par Microsoft. Il a également fallu installer le plugin Flash développé par Adobe (anciennement Macromedia) pour Pocket PC. Une fois ces différentes opérations réalisées, nous avons pu naviguer sur Internet et en particulier sur le site de la Nièvre. La page d'accueil du site, comme nous pouvons nous en rendre compte à la figure 4.1 page suivante, la page est totalement illisible.

---

<sup>1</sup>[www.nievre-tourisme.com](http://www.nievre-tourisme.com)



Figure 4.1 – Page d'accueil du site de la Nièvre

Après quelques recherches et une étude un peu plus approfondie du code source de la page, nous nous sommes rendus compte que le problème d'affichage provenait de la manière dont la page a été écrite, et plus particulièrement l'utilisation de frames non supportées par notre navigateur web. Nous reviendrons plus en détails dans la section 6 page 59 mais l'utilisation de frames est fortement déconseillée voir même « prohibée » par le W3C et les autres organismes travaillant à l'amélioration des sites web pour les personnes atteintes de handicaps.

Si nous passons la page d'accueil pour se rendre directement à la page consacrée aux recherches et aux offres touristiques, nous pouvons voir sur la figure 4.2, figure 4.3 et sur la figure 4.4 que, bien qu'inadapté pour les périphériques mobiles, les pages sont lisibles. Ainsi, au vu de ces résultats, nous pouvons dire que nous devons adapter l'affichage actuel pour les périphériques mobiles, sans devoir tout redévelopper depuis le début. Nous ferons des propositions d'adaptations et des méthodes possibles pour y arriver, plus loin, dans ce mémoire.



Figure 4.2 – Recherche en plein texte



Figure 4.3 – Recherche en utilisant l'ontologie



Figure 4.4 – Historique des étapes de la recherche

L’affichage des fiches des objets se fait également correctement comme en témoigne la figure 4.5 à l’exception de la rubrique « Vous aimerez aussi » (trois pancartes de bois avec des suggestions à l’intérieur) où le texte se superpose sur l’image et le tout est totalement illisible comme on le voit sur la figure 4.6. Enfin, l’affichage de la carte géographique issue de Google Maps ne fonctionne pas. Il faudra donc voir comment l’API de Google Maps fonctionne et si il est possible de le faire fonctionner sur des PDA. En effet, à ce moment de l’analyse, il nous est impossible de dire si le problème provient de l’application Google Maps qui est incompatible avec les périphériques mobiles ou bien si le problème vient du site de la Nièvre et le code utilisé pour l’affichage de la carte. Il faudra donc s’intéresser plus particulièrement à l’adaptation de l’affichage et régler le problème de la carte géographique et des suggestions par l’intermédiaire des pancartes en bois.



Figure 4.5 – Affichage de la fiche d’un objet



Figure 4.6 – Problème d’affichage des suggestions

Maintenant, nous allons nous intéresser à l'existant du tourisme sur Internet (ViaMichelin par exemple) ainsi que les guides touristiques classiques (guide vert Michelin, le guide du routard, etc.).

## 4.2 Analyse du domaine du tourisme

### 4.2.1 Le tourisme sur Internet

Le domaine touristique est un sujet largement représenté sur Internet sous différentes formes. On trouve bien évidemment les sites d'agences de voyages, les sites des compagnies aériennes pour les avions ou les compagnies ferroviaires pour le train, les sites de réservations et d'achats de billets, etc. mais également les sites de restaurants, hôtels, offices de tourisms, etc. Ce sont surtout les sites du deuxième type qui nous intéresseront dans notre travail. Pour cela, nous allons analyser différents sites web pour nous permettre d'avoir différentes idées dont nous pourrions nous inspirer.

#### ViaMichelin

Les services offerts par le site ViaMichelin<sup>2</sup> sont assez vastes et variés : cartes et plan, itinéraires, hôtels, tourisme, automobile, navigation GPS mais seuls les rubriques tourisme et hôtels nous intéressent. Les deux rubriques sont assez semblables dans le fonctionnement de la recherche tout du moins. En effet, pour la rubrique tourisme, on propose surtout de mettre en valeur périodiquement certaines villes ou régions.

Au premier abord, la recherche d'information n'est pas des plus intuitives et le contenu des fiches assez pauvres et la structuration des données n'est pas approfondie. On retrouve en effet le nom du lieu, une indication concernant l'intérêt plus ou moins grand du site, son type (château, site naturel, attraction/-loisir, etc.), ses coordonnées (adresse, téléphone, ...), les conditions d'ouvertures sous forme de texte, le tarif et une courte description et une petite carte pour situer le site dans le quartier où il se trouve (l'échelle est petite et donc on situe uniquement les quelques rues aux alentours seulement). Rien de plus et avec une présentation classique. La conclusion de l'analyse du site est donc assez mitigée. Ce qu'il faut principalement retenir du contenu du site de Michelin, ce sont les critères de recherches sur lesquels nous reviendrons dans la section 5.1.2 page 39.

#### Site de l'office du tourisme de la Nouvelle-Zélande

Le site touristique de la Nouvelle-Zélande<sup>3</sup> et son *Travel Planner* permet à l'utilisateur de mémoriser les pages du site qu'il trouve intéressantes, les activités, les logements, etc. et ensuite des les situer sur la carte interactive de la Nouvelle-Zélande, de planifier ces différents éléments sur un calendrier afin d'organiser son futur séjour. Le système Travel Planner permet également de fournir des informations comme la distance entre deux activités, les moyens et le temps de transports pour s'y rendre. Une fois qu'il est content du séjour qu'il a planifié, l'utilisateur peut le soumettre à des agences de voyages qui se chargera ensuite des différentes réservations. On peut sauvegarder les éléments du Travel Planner pour y revenir par la suite et y apporter des modifications. Après avoir analysé plus en détails le site, une amélioration possible serait la validation du séjour planifié par l'utilisateur. En effet, sur le site, on trouve les horaires et jours d'ouverture/fermeture des activités et si on planifie un élément à un jour incorrect, le système ne nous prévient pas de l'erreur. De même, on peut sélectionner deux éléments séparés par plusieurs centaines voire plus d'un millier de kilomètres de distance dans la même journée et donc on pourrait apporter des fonctionnalités configurables pour avertir de l'utilisateur de ce genre de remarques. L'inconvénient majeur du site, dans sa volonté d'interactivité et de dynamisme, est sa fluidité à cause de l'utilisation fréquente de la technologie Flash d'Adobe pour le Travel Planner et la carte interactive. Ce genre de procédé poserait quelques problèmes sans adaptation pour les périphériques mobiles due à la puissance réduite de calcul et l'utilisation de bande passante réduite (connexion à 56k en général et plus

---

<sup>2</sup>[www.viamichelin.fr](http://www.viamichelin.fr)

<sup>3</sup><http://www.newzealand.com/travel>



rapide avec l'UMTS ou ses dérivés mais encore peu répandu chez les utilisateurs ordinateurs lorsque des bornes wifi ne sont pas disponibles).

En conclusion, le portail touristique de la Nouvelle-Zélande est très moderne avec des fonctions avancées dans la planification de son séjour et la localisation des objets sur une carte interactive. Du côté du contenu touristique, nous ne trouvons pas plus d'informations que par exemple sur le site de Michelin et la recherche d'activité se fait thématiquement mais de façon relativement élémentaire au vu des options proposées.

## MadeinMtl

Le site *MadeinMtl* est un site éditorial fournissant des informations touristiques sur la ville de Montréal mais il tient compte de de l'avis des utilisateurs et de leur manière de naviguer. En effet, par l'intermédiaire « d'arrêts » (qui correspondent aux différents objets disponibles : restaurants, boutiques, musées, etc.), l'utilisateur peut classer les arrêts par leur date de publication, leur cotation ou bien en fonction du nombre de fois qu'ils ont été envoyés par mail (pour faire découvrir l'objet à un ami). De plus, la manière d'aborder la navigation est intuitive et originale : il y a le type de recherche classique par thématiques (manger, apprendre, acheter, dormir, ...) mais il est également possible d'explorer la ville par l'intermédiaire d'une carte et on peut afficher le type d'arrêt de son choix dessus. Il suffit ensuite de cliquer sur le signet pour obtenir tous les détails de l'élément.

Concernant la recherche, comme nous l'avons indiqué précédemment, il y a un système de menu déroulant à 3 niveaux. Ainsi, on spécifie ce que l'on souhaite (manger, dormir, acheter, danser, se détendre, etc.) et ensuite, en fonction du premier choix, les choix disponibles pour le 2<sup>ème</sup> niveau permettent d'affiner les domaines de recherche et idem pour le dernier niveau. Sinon, lorsque l'on est sur la fiche d'un objet, il y a une option de recherche permettant de trouver des arrêts du même type que l'actuel avec en plus la possibilité de sélectionner plus finement le type d'objet souhaité.

Le site propose également des parcours thématiques dans Montréal ainsi qu'un calendrier des événements. Ce sont de bonnes idées qu'il peut être intéressant de retenir pour le futur projet. Concernant l'affichage des résultats, on a un affichage concis avec simplement le titre, le pourcentage d'adéquation du résultat par rapport à la recherche, une icône pour identifier la catégorie et une note pour indiquer l'opinion des utilisateurs sur l'item affiché. En dessous de la liste des résultats, une carte permet de localiser chacun des items sur un plan de la ville.

Enfin, comme beaucoup de sites le permettent maintenant, le site *MadeinMtl* permet d'enregistrer dans son compte, les éléments que l'on souhaite retrouver lors d'une prochaine visite.

## 4.2.2 Les guides touristiques

Toujours très utilisés par les touristes, surtout lorsqu'ils sont sur le lieu de leur séjour (la majorité des touristes ne possèdent pas encore d'un PDA ou autre périphérique mobile) afin de planifier leurs activités et surtout trouver les informations historiques sur les visites, les coordonnées des restaurants ou hôtels, etc.

Ainsi, afin de continuer notre exploration du domaine touristique, nous nous sommes intéressés aux deux principaux guides que sont *le guide vert Michelin* et *le guide du routard*. Il en existe bien d'autres mais malheureusement nous ne pouvions pas nous permettre de tous les analyser, surtout qu'il ne s'agit là que d'une phase préliminaire à notre travail de recherche.

Le guide du routard analysé date de quelques années donc les choses ont pu évoluer et s'être améliorées depuis. Sinon il n'y guère d'idées de présentation ou d'information à reprendre en particulier car la présentation est finalement assez austère (en noir et blanc sans photo), l'information classique et sans originalité. La particularité du guide du routard se trouve être finalement la richesse des adresses que l'on retrouve à l'intérieur (les brasseries ou gîtes locaux avec un rapport qualité/prix irréprochable). Le guide Michelin est beaucoup plus attractif car en couleurs, avec des photos et une présentation beaucoup plus aérée et lisible et s'adresse plutôt à un autre public que le guide du routard. En effet, on trouve en plus grand nombre des hôtels et restaurants de plus haut standing que celui du guide du routard. Pour conclure, on peut se trouver un peu déçu de voir l'existant si figé dans le domaine touristique et que

finally, les choses ont très peu évolué depuis des dizaines d'années si on compare deux versions du guide Michelin. Cela nous laisse néanmoins une plus grande marge de manoeuvre dans notre travail afin d'améliorer les choses et de proposer un contenu beaucoup plus riche.

### 4.2.3 Autres supports

Même si d'après nos analyses précédentes du web et des guides touristiques classiques nous avons pu constater un certain manque d'avancées, l'arrivée de nouveaux « gadgets » technologiques comme les lecteurs MP3 (iPod, Archos, etc. qui sont maintenant multimédia), les nouvelles générations de consoles portables (Sony PSP, Nintendo DS) et évidemment les PDA permettent d'entrevoir un grand potentiel pour le domaine du tourisme. En effet, les touristes les ont toujours avec eux et il est possible d'y stocker du contenu multimédia et donc pourquoi pas du contenu touristique ? Ainsi, il serait possible d'utiliser les lecteurs MP3 comme audio-guides (ainsi on pourrait se passer des audio-guides que l'on peut actuellement louer lors des visites de musées ou autre monument en proposant simplement le contenu audio au touriste qui le stockerait dans son lecteur MP3 par exemple). De plus, les nouveaux modèles de lecteurs MP3, en plus de pouvoir lire de la musique, permettent de lire des vidéos d'assez bonne qualité et donc nous pourrions, en plus du contenu sonore, offrir un contenu vidéo aux touristes pour agrémenter ses visites. On retrouve déjà quelques essais dans ce domaine pour le château de Versailles (on peut télécharger gratuitement des Podcasts afin d'obtenir quelques informations historiques sur les jardins du château, le Trianon, ...). Il serait donc vraiment intéressant de se pencher vers ce genre de possibilité pour faire entrer le tourisme dans le 21<sup>ème</sup> siècle. Ce qui est possible pour les lecteurs MP3 l'est également pour les PDA puisque depuis plus longtemps encore, il est possible d'écouter de la musique et de regarder des vidéos sur ce type de périphériques qui a l'avantage de posséder un plus grand écran et sur un grand nombre une connexion wifi et bluetooth permettant une plus grande possibilité d'interactivité avec l'utilisateur. Concernant enfin l'iPod, depuis quelques temps maintenant, il est possible d'acheter des jeux. Même si cela ne nous intéresse pas directement, il est utile de savoir qu'il est possible de développer des applications pour ce périphérique et donc pourquoi pas des applications touristiques et une visite virtuelle du musée comme cela existe déjà sur CD-Rom ou DVD-Rom pour les ordinateurs de bureaux. L'inconvénient est que le type d'application est propriétaire à Apple et donc il est nécessaire de passer des accords avec ce dernier si l'on souhaite s'orienter vers cette possibilité. Néanmoins, on sait que cette éventualité existe.

Sony a vu tout le potentiel et la forte demande du tourisme dans le domaine de la mobilité. Ainsi, ils ont développé tout une série de « logiciels » pour sa console portable, la PSP, appelé *Passport to* pour les grandes capitales européennes (Paris, Londres, Amsterdam, Prague, ...). On retrouve dans le logiciel aussi bien un contenu textuel classique d'informations sur les monuments mais également un contenu multimédia avec des vidéos, des commentaires audio, des cartes interactives de la ville (plan, métro, ...) afin de pouvoir se déplacer mais également des itinéraires thématiques de ballades, une liste de restaurants avec leurs coordonnées et un résumé de ce que l'on trouve sur la carte, etc. Grâce à la petite taille de l'appareil, il est aisé d'emmener la PSP avec soi lors de ses déplacements et à la taille de son écran, il est vraiment facile de la transformer, grâce au logiciel, en guide touristique. De plus, Sony a développé un autre logiciel, appelé *Talkman*, ayant comme fonctionnalité d'apprendre des langues étrangères. Vendu avec un micro, il permet de perfectionner sa prononciation et il possède un grand nombre de phrase-type utiles lorsque l'on se rend dans un pays étranger pour se débrouiller dans le quotidien. Avec ses deux logiciels en poche, n'importe qui peut donc se transformer en parfait globe-trotter.

### 4.2.4 Conclusion

Après notre analyse du domaine touristique et ce qui existe actuellement, on se rend compte qu'il y a encore du travail et des améliorations à apporter mais que l'arrivée de nouveaux périphériques très en vogue auprès des jeunes pourraient être une solution à ces problèmes. En effet, le multimédia fait maintenant parti de notre quotidien (appareil photo et caméscope dans les nouveaux GSM par exemple) et les éditeurs l'ont bien compris. Il est donc nécessaire d'en tenir compte si l'on souhaite profiter de ce

marché énorme en terme financier qu'est le tourisme.

Nous allons maintenant nous intéresser au contenu qu'il est important d'avoir pour les futures fiches touristiques afin de pallier les manques actuels. Après cela, il sera possible de nous pencher sur la manière de représenter et de visualiser cette information sur les périphériques mobiles.



---

## Analyse et structuration des fiches touristiques

---

### 5.1 Structure des fiches touristiques existantes

Après avoir dressé un bilan de ce qui existait sur Internet et d'autres supports, nous allons maintenant nous intéresser dans ce chapitre à la structuration de l'information que l'on souhaitera proposer aux utilisateurs. En effet, si nous souhaitons utiliser les dernières technologies et avancées des recherches dans le domaine des systèmes d'information, nous devons avoir une information structurée et possédant une sémantique claire et sans ambiguïté.

Dans un premier temps, nous allons extraire les différents éléments constituant les fiches du site de la Nièvre et de Michelin ainsi que les critères de recherche offerts par ces derniers. Par la suite, nous développerons notre propre contenu et avec notre propre structure afin de pouvoir partir sur de bonnes bases pour la représentation et la visualisation de ces informations sur des périphériques mobiles.

#### 5.1.1 Contenu des fiches du site de la Nièvre

Le site touristique de la Nièvre est au début de son exploitation et de sa mise à disposition du public. Il est en version 1.0 et donc il reste encore beaucoup de choses à perfectionner ou implémenter. Actuellement, l'élément le plus avancé du site est le moteur de recherche qui exploite l'ontologie touristique. L'affichage des résultats est classique (à la manière de Google, avec quand elle est disponible, une photo de l'hébergement ou bien de l'activité proposée mais pour le moment, il n'y en a pas beaucoup) et la fiche de l'objet est sommaire, comme on peut le voir sur les deux captures de la figure ?? page ?? et de la figure ?? page ?. Le contenu est vraiment très pauvre et l'utilisateur, au final, ne possède pas beaucoup plus d'informations qu'avant d'effectuer sa recherche et il doit consulter le site internet de l'objet s'il est mentionné dans la fiche. L'utilité est donc assez restreinte. Nous allons donc maintenant nous baser sur notre analyse précédente de l'existant du tourisme pour définir un ensemble d'informations utiles que nous souhaiterions renseigner aux utilisateurs.

#### 5.1.2 Contenu des fiches du site Via Michelin

Nous ne reviendrons pas sur les commentaires que nous avons fait dans la section 4.2.1 page 34. Dans cette section, nous allons présenter les informations que l'on peut trouver dans les fiches des différents types d'objets sur le site de Michelin afin de nous en inspirer pour établir le contenu des fiches des objets touristiques que nous souhaitons développer.

Pour un lieu touristique, nous trouvons les informations générales le concernant tel que son nom, une

indication concernant l'intérêt plus ou moins grand du site, son type (château, site naturel, attraction/-loisir, etc...), ses coordonnées (adresse, téléphone, ...), les conditions d'ouvertures sous forme de texte, le tarif et une courte description. On trouve également une petite carte pour situer le site dans le quartier où il se trouve (l'échelle est petite et donc on situe uniquement les quelques rues aux alentours).

Le procédé de recherche sur le site est classique avec le pays, l'adresse, le code postal et la ville. On peut également affiner la recherche suivant l'intérêt touristique avec les critères suivants :

- vaut le voyage
- mérite un détour
- intéressant

et suivant les thèmes :

- architecture civile et historique
- édifices religieux
- musées
- rues et quartiers historiques
- sites urbains, sites naturels
- attractions et loisirs
- châteaux

Pour chaque critère, une icône lui est associée. Plus que le contenu des fiches qui est classique et sans information « originale » et nouvelle par rapport à ce qui existe ailleurs, ce sont les différents critères qu'il faut retenir pour notre travail.

De même, il existe sur le site de Michelin, un moteur de recherche pour les hôtels ainsi que pour les restaurants. Nous avons pu donc, comme pour les lieux et activités touristiques, extraire les critères suivants pour les hôtels :

- présenté dans le guide Michelin
- restaurant
- air conditionné
- piscine
- internet
- TV satellite
- parking
- accès handicapé
- chambres non fumeur
- animaux autorisés
- enfants.

De même, pour les restaurants, nous avons pu extraire les critères suivants :

- les exclusifs du guide (Bib gourmand, restaurant étoilé, agréable/tranquille)
- catégorie (grand luxe, grand confort, très confortable, de bon confort, assez confortable, Pubs/Tapas)
- menu enfant
- menu économique
- parking
- installations accessibles aux personnes handicapées physiques.

## 5.2 Réflexions sur le contenu des fiches touristiques

En nous basant sur les différents éléments précédemment exposés, nous avons quelques idées de départ à affiner pour le contenu des fiches des objets. Par la suite, il faudra ordonner le contenu afin de pouvoir ensuite définir une structure DTD des fiches XML.

Voici les éléments retenus pour une fiche :

- nom
- type parmi une liste pré-défini par rapport à l'ontologie de Mondeca (hébergement, activité, restaurant, ...)
- sous-type parmi une liste qui dépend du type :
  - hébergement → hôtel, camping, gîte, ...
  - restaurant → rapide, gastronomique, ...
  - activité → sportive, culturelle, éducative, ...
  - ...
- note de l'objet suivant son intérêt
- prix (ou au moins une fourchette de prix pour certains cas)
- adresse complète
- téléphone, fax
- email
- site internet
- description
- coordonnées géographiques pour situer le résultat sur une carte ou établir un itinéraire
- photo
- dates d'ouverture
- heures d'ouvertures
- langue parlée par le personnel ou les employés

Ensuite, il y a des éléments qui dépendent du type de l'objet. Ainsi pour les activités, on indiquera une durée approximative de la visite ou de l'activité ainsi que le public visé.

Pour les hébergements, il faudra ajouter les éléments suivants :

- type de standing
- nombres d'étoiles ou équivalents
- labels possédés (à définir à partir de l'ontologie de la Nièvre)
- répartition des chambres par type (2 pers., 3 pers., suite, avec s-d-b, avec douche, ...)
- caractéristiques supplémentaires basées sur l'existant

Pour les restaurants, il faudra ajouter les éléments suivants :

- type de standing
- type de gastronomie (française, italienne, japonaise, ...)
- nombre d'étoiles, fourchettes ou équivalents
- labels éventuels (recommandé par le guide du routard, ...)
- caractéristiques supplémentaires basées sur l'existant
- prix (même si l'élément porte le même nom que pour les autres objets, la sémantique est différente ici)
- périodes d'ouvertures (même remarque que pour le prix)

### 5.2.1 Suggestions supplémentaires

Maintenant que les éléments souhaités sont énumérés, voici quelques idées de comportements que l'on pourrait adopter pour la fiche. Par exemple, on pourrait, lorsqu'il s'agit d'une activité, suggérer un restaurant et un logement proche, dans la même catégorie de standing (par exemple, si la personne choisit l'activité de golf, on évitera de lui proposer un camping et un établissement de restauration rapide mais

plutôt un restaurant et hôtel dans une gamme de prix plus élevée. Il faudra, comme cela existe pour le site de la Nièvre mais qui ne fonctionne pas sur PDA, afficher une carte de la région pour situer l'objet. On pourrait aussi par exemple donner la distance kilométrique qui sépare les différents objets si on décide d'implémenter un panier afin de donner une idée aux utilisateurs.

Il pourrait être aussi intéressant de proposer des circuits touristiques aux utilisateurs en fonction des activités qu'il a choisi.

On pourrait également donner les coordonnées (ou l'adresse du site internet s'il existe) de l'office du tourisme rattaché à l'objet sélectionné si l'utilisateur veut obtenir de plus amples informations.

Maintenant que nous avons exposé les différents éléments que nous voulions inclure dans la fiche de contenu, nous allons nous intéresser à la formalisation de cela sous forme d'un schéma entité-relation afin de clarifier les idées et pouvoir ensuite en déduire la structure physique des fiches.

### 5.3 Schéma entité-relation des fiches

Maintenant que nous avons précisé le type d'information que nous souhaitons développer pour les fiches touristiques, schématisons cela graphiquement afin de rendre cela plus compréhensible. Dans ce but, nous allons utiliser des schémas entité-relation et pour nous aider dans notre tâche, l'outil CASE développé par une équipe des facultés de Namur (FUNDP), Db-Main<sup>1</sup>.

Pour commencer, voici la version 1 de notre schéma (figure 5.1 page ci-contre) que nous expliciterons et corrigerons/améliorerons par la suite :

---

<sup>1</sup>[www.db-main.be](http://www.db-main.be)



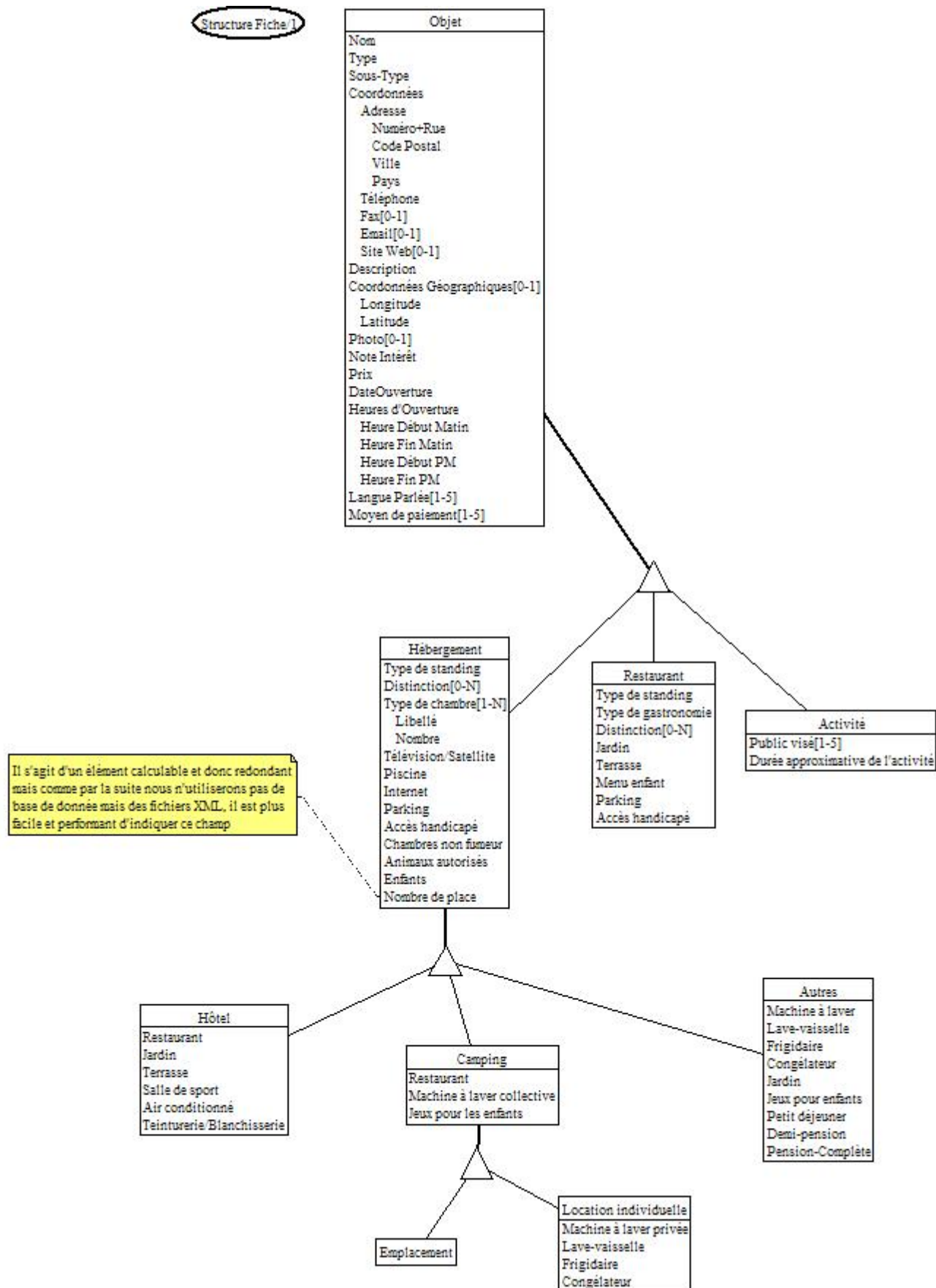


Figure 5.1 – Version 1 du schéma ERA de la structure des fiches

Après réflexion, nous avons vu que la manière de stocker les dates et heures d'ouverture n'était pas bonne car il est fréquent d'avoir, pour les activités touristiques saisonnières, différents horaires et jours d'ouverture en fonction de la saison. Ainsi, une activité touristique est ouverte plus longtemps (que ce soit en terme de nombre de jour dans la semaine mais également les horaires chaque jour) en période estivale que durant l'hiver surtout s'il s'agit d'une activité d'extérieur. Nous avons également le cas des activités de sport d'hiver qui sont fermées l'été, etc. Nous avons donc décidé d'approfondir notre réflexion sur ce point afin d'avoir une structure plus flexible et permettant un meilleur stockage de l'information et un traitement plus efficace par la suite.

Nous en sommes arrivés à la conclusion qu'il était plus pratique dans notre cas, que nous manipulerions des périodes horaires composée des éléments suivants :

- d'une date de début de période
  - d'une date de fin de période
  - d'un tableau d'ouverture de 7 éléments (un élément pour chaque jour de la semaine) composé :
    - d'un booléen pour indiquer si l'objet est ouvert ce jour de la semaine
    - d'une heure d'ouverture pour le matin
    - d'une heure de fermeture pour le midi
    - d'une heure d'ouverture pour l'après-midi
    - d'une heure de fermeture pour le soir
- les 4 champs pour les heures vaudront 0 lorsque le booléen est égal à false (indiquant que l'objet est fermé)

Nous aurons ainsi d'une à N périodes horaire nous permettant d'avoir une structure assez simple tout en pouvant gérer tous les cas possibles concernant les jours et heures d'ouverture.

Du même style, nous devons revoir notre structure de stockage des prix, tout du moins pour les activités. En effet, pour les restaurants et les hébergements il est impossible de donner l'ensemble des détails de chaque prix, pour chaque période et pour chaque chambre ou logement (il ne faut pas oublier qu'on est dans un contexte touristique où l'on souhaite informer les utilisateurs de l'offre proposée par la région mais nullement de faire un système de réservation et gestion d'un hébergement) et nous nous contenterons d'une fourchette de prix entre le minimum et le maximum de l'établissement mais concernant les activités, les choses sont différentes. Nous avons un prix fixe, parfois par période (le prix est plus élevé lors de la pleine période touristique) et moins élevé le reste du temps mais nous avons également différents tarifs en fonction du critère de l'âge ou de groupe. Il est donc intéressant de pouvoir intégrer ce genre d'information dans notre fiche. Pour cela, le prix, au lieu de se trouver dans l'entité "Objet", se retrouvera au niveau inférieur, c'est-à-dire au niveau de "Hébergement", "Restaurant" et "Activité". Pour les deux premiers, il s'agira donc d'une fourchette (intervalle) tandis que pour le dernier, "Activité", il s'agira d'un groupe d'attribut "période de prix" comme pour les périodes d'ouverture précédemment (il pourra y avoir de une à N périodes) dont voici le détail :

- d'une date de début de période
- d'une date de fin de période
- d'un tarif enfant
- d'un tarif adulte
- d'un tarif de groupe
- d'un tarif réduit.

En continuant à observer et analyser le schéma, nous avons également pensé à une autre structure pour le prix, plus simple, qui serait composée d'un groupe de 2 attributs donc voici le détail :

- libellé de l'âge (enfant, adulte, groupe, tarif réduit)
- prix.

Dans les deux cas, nous avons décidé de nous restreindre à seulement 4 tarifs différents mais on pourrait ensuite l'adapter à plus si le besoin s'en faisait ressortir. Mais dans un premier temps, nous pensons que cela est largement suffisant pour la majorité des cas possibles, surtout que nous travaillons pour le moment sur un cas expérimental afin de pouvoir réfléchir sur la manière de visualiser cet ensemble d'information. Nous avons donc décidé de retenir la deuxième solution : plus simple à mettre en place et

permettant d'être plus proche de la réalité.

En effet, il n'est pas fréquent, de trouver des tarifs différents suivant la période, pour des activités. Cela est courant pour les hébergements (avec la pleine saison et le hors-saison) mais pour ces objets, nous travaillons avec un système de fourchette de prix donc nous n'en avons pas réellement besoin. De plus, comme nous en avons détaillé la structure, il serait possible, en cas de demande des futurs utilisateurs (offices du tourisme, organismes régionaux chargés du tourisme, ...), de le mettre en place en remplacement de la représentation que nous avons choisie.

Voici maintenant le nouveau schéma ERA (figure 5.2 page suivante) après les modifications dont nous venons de discuter :

Maintenant que nous avons défini de manière assez détaillée la structure de nos différentes fiches d'objets touristiques, nous allons devoir porter notre réflexion sur la manière de stocker le contenu des fiches.

## 5.4 Stockage physique du contenu des fiches

Différentes technologies s'offrent à nous pour stocker le contenu des fiches et ensuite développer notre portail touristique dessus. En effet, nous pourrions utiliser une base de donnée, des fichiers traditionnels, des fichiers XML, etc. On peut exclure d'emblée les fichiers traditionnels car nous ne pourrions pas ajouter de la sémantique aux informations facilement et puis une modification global nécessiterait la modification dans chaque fichier, un par un. Il nous reste donc deux systèmes possibles et tout deux compatibles avec le monde du web.

Après une réunion avec l'équipe de Mondeca et d'autres partenaires du projet, on nous présente plus en détails le fonctionnement du système et plus particulièrement des deux outils développés par Mondeca, **ITM** (qui gère l'ontologie, le front-office et le back-office du site web) et par Antidot, **AFS** (qui gère l'indexation du contenu texte), nous avons constaté que l'ensemble du système tournait autour de fichiers XML. Notre interrogation concernant le type de stockage des fiches est terminée et nous avons donc décidé de nous orienter vers ce même chemin pour des raisons de compatibilité et puis également en cas de questions ou de problèmes, nous pourrions profiter des conseils des différents partenaires. Vous pouvez trouver en annexe, le code source d'un fichier XML du site de la Nièvre. Cela nous permet ainsi de comprendre l'intégration de l'ontologie dans les fiches touristiques et la manière dont est structurée l'information. De nouveau, après l'étude de ces fichiers, nous avons le sentiment d'un manque de structure et donc cela est un frein à l'évolution des fonctionnalités que l'on pourrait offrir aux utilisateurs sur le portail touristique. Cela nous conforte donc dans notre choix de partir de zéro et de définir notre propre structure de l'information.

Maintenant que nous sommes fixés sur la manière de stocker les fiches, nous pouvons travailler sur la structure physique de ces dernières.

## 5.5 Structure physique des fiches touristiques

### 5.5.1 Analyse de la structure physique des fiches touristiques

Maintenant que nous sommes fixés sur le contenu de l'information que nous souhaitons stocker dans les fiches XML, il nous reste à définir et réfléchir sur la manière dont nous allons stocker cette information concrètement dans le fichier XML.

En effet, si nous souhaitons utiliser des techniques sémantiques pour la navigation et la recherche sur le portail touristique, nous devons structurer notre information. Pour cela, nous devons nous baser sur le schéma entité-relation de la figure 5.2 page suivante et stocker les données dans des fichiers XML de manière structurée. Pour cela, nous allons définir cette structure en utilisant une DTD<sup>2</sup>. Ce « langage

---

<sup>2</sup>Document Type Definition

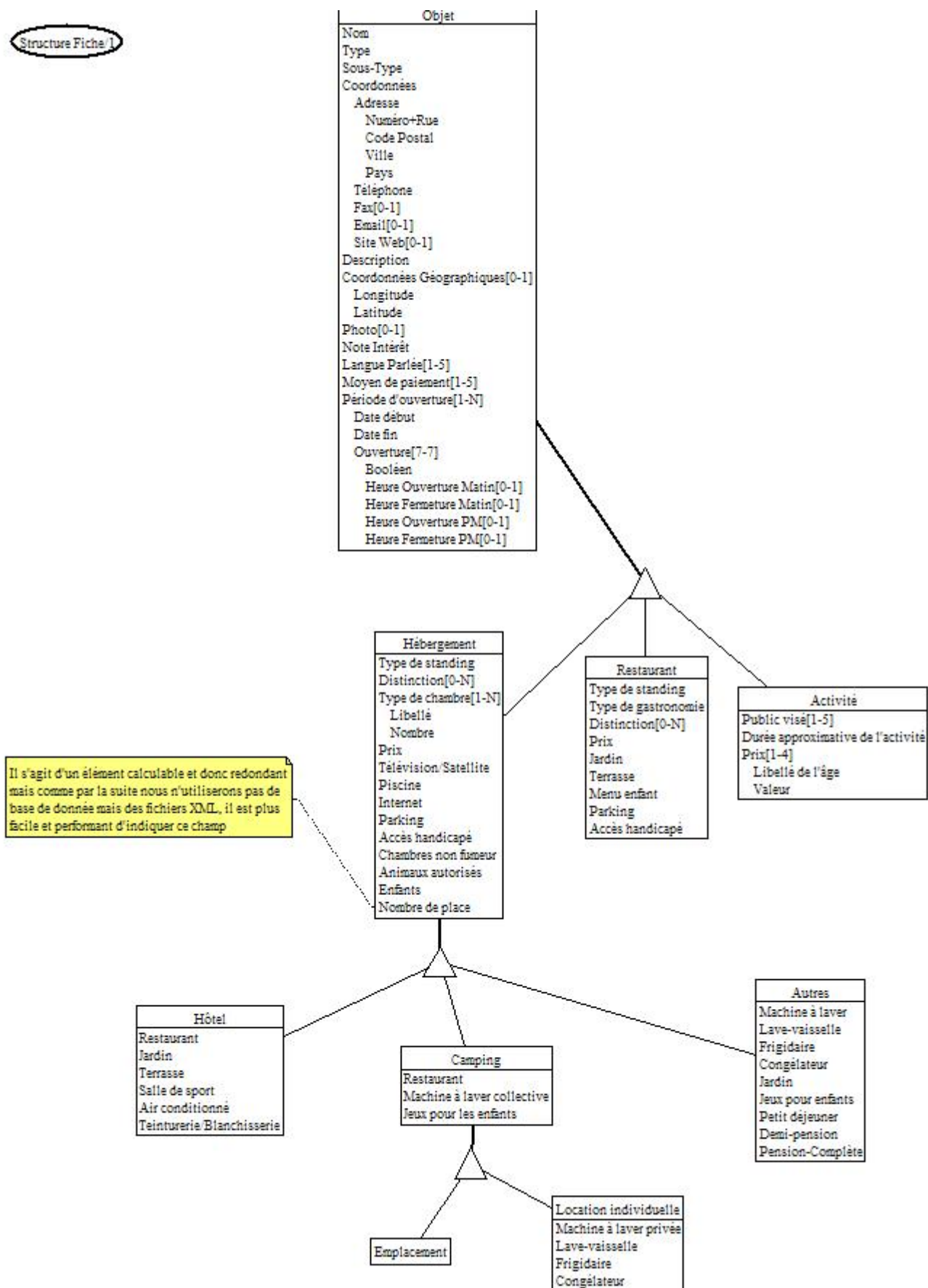


Figure 5.2 – Version 2 du schéma ERA de la structure des fiches

» permet de définir les différents éléments qui doivent composer le fichier XML, leur ordre, leur nombre (obligatoire, facultatif, un seul, plusieurs, ...). Dans une DTD, nous ne pouvons pas indiquer le type de variable de chaque élément. Un élément peut être simplement un ensemble de plusieurs éléments ou bien un *#PCDATA* qui correspond à une chaîne de caractères. On peut également définir des entités mais nous ne nous sommes pas attardés sur ce sujet car après avoir survolé le sujet, nous avons pu nous rendre compte que nous n'en avions pas besoin pour notre travail. Il est donc impossible de définir des types du genre date, entier, des décimaux, etc... Cela est par contre possible dans un XSD (XML Schema Definition) mais la liste des types variables n'est néanmoins pas beaucoup plus étendu ou bien trop éloignée de nos besoins.

De toute manière, étant donné que nous n'avons pas de calcul à faire sur des nombres, des dates, etc... nous pensons que le plus simple sera d'utiliser des chaînes de caractères pour tous les éléments en définissant une syntaxe et une sémantique formelle pour chaque élément et ainsi, il nous suffira de parcourir le fichier XML pour en extraire l'information souhaitée.

Voici le code source du fichier DTD :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
  <!ELEMENT objet (hotel | emplacement | loc_individuelle | autres |
    restaurant | activite)>
  <!ELEMENT hotel (hebergement, rest_bool, jardin_bool, terrasse_bool,
    salle_sport_bool, clim_bool, blanchisserie_bool)>
  <!ELEMENT emplacement (camping, superficie?)>
  <!ELEMENT loc_individuelle (camping, machine_laver_privée_bool,
    lave_vaisselle_bool, frigo_bool, congelateur_bool)>
  <!ELEMENT camping (hebergement, rest_bool, machine_laver_collective_bool,
    jeux_enfants_bool)>
  <!ELEMENT autres (hebergement, machine_laver_privée_bool,
    lave_vaisselle_bool, frigo_bool, congelateur_bool, jardin_bool,
    jeux_enfants_bool, petit_dej_bool, demi_pension_bool,
    pension_compl_bool)>
  <!ELEMENT hebergement (info, type_standing, distinction*, type_chambre+,
    prix, tv_satellite, piscine_bool, internet_bool, parking_bool,
    acces_hand_bool, non_fumeur_bool, animaux_bool, enfants_bool,
    nbr_place)>
  <!ATTLIST hebergement soustype (hotel | camping | mobilhome |
    chambre_hote | gite | location) #REQUIRED>
  <!ELEMENT restaurant (info, type_standing, type_gastronomie, distinction
    *, prix, jardin_bool, terrasse_bool, menu_enfant_bool, parking_bool,
    acces_hand_bool)>
  <!ATTLIST restaurant soustype (gastronomique | brasserie | rapide) #
    REQUIRED>
  <!ELEMENT activite (info, public_vise*, duree_activite, prix_activite+)>
  <!ATTLIST activite soustype (sportive | culturelle | éducatrice) #
    REQUIRED>
  <!ELEMENT info (nom, coordonnees, description, coordgeo?, photo?,
    noteinteret?, langueparlee+, paiement+, periodeouverture+)>
  <!ELEMENT coordonnees (adresse, telephone, fax?, email?, url?)>
  <!ELEMENT adresse (numero?, rue, code_postal, ville, pays)>
  <!ELEMENT coordgeo (longitude, latitude)>
  <!ELEMENT periodeouverture (datedebut, datefin, ouverture*)>
  <!ELEMENT ouverture (ouvert_bool, heure_ouv_matin?, heure_ferm_matin?,
    heure_ouv_PM?, heure_ferm_PM?)>
  <!ELEMENT type_chambre (libelle, nombre)>
  <!ELEMENT prix_activite (libelle_age, valeur)>
  <!ELEMENT rest_bool (#PCDATA)>
  <!ELEMENT jardin_bool (#PCDATA)>
  <!ELEMENT terrasse_bool (#PCDATA)>
  <!ELEMENT terrasses_bool (#PCDATA)>
  <!ELEMENT salle_sport_bool (#PCDATA)>
  <!ELEMENT clim_bool (#PCDATA)>
  <!ELEMENT blanchisserie_bool (#PCDATA)>
  <!ELEMENT superficie (#PCDATA)>
  <!ELEMENT machine_laver_privée_bool (#PCDATA)>
  <!ELEMENT lave_vaisselle_bool (#PCDATA)>
  <!ELEMENT frigo_bool (#PCDATA)>
  <!ELEMENT congelateur_bool (#PCDATA)>
  <!ELEMENT machine_laver_collective_bool (#PCDATA)>
```

```

<!ELEMENT jeux_enfants_bool (#PCDATA)>
<!ELEMENT petit_dej_bool (#PCDATA)>
<!ELEMENT demi_pension_bool (#PCDATA)>
<!ELEMENT pension_compl_bool (#PCDATA)>
<!ELEMENT type_standing (#PCDATA)>
<!ELEMENT type_gastronomie (#PCDATA)>
<!ELEMENT distinction (#PCDATA)>
<!ELEMENT prix (#PCDATA)>
<!ELEMENT tv_satellite (prise_bool?, television_fournit_bool)>
<!ELEMENT prise_bool (#PCDATA)>
<!ELEMENT television_fournit_bool (#PCDATA)>
<!ELEMENT piscine_bool (#PCDATA)>
<!ELEMENT internet_bool (#PCDATA)>
<!ELEMENT parking_bool (#PCDATA)>
<!ELEMENT acces_hand_bool (#PCDATA)>
<!ELEMENT non_fumeur_bool (#PCDATA)>
<!ELEMENT animaux_bool (#PCDATA)>
<!ELEMENT enfants_bool (#PCDATA)>
<!ELEMENT nbr_place (#PCDATA)>
<!ELEMENT public_vise (#PCDATA)>
<!ELEMENT duree_activite (heure, minute)>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT photo (#PCDATA)>
<!ELEMENT noteinteret (#PCDATA)>
<!ELEMENT langueparlee (#PCDATA)>
<!ELEMENT paiement (#PCDATA)>
<!ELEMENT numero (#PCDATA)>
<!ELEMENT rue (#PCDATA)>
<!ELEMENT code_postal (#PCDATA)>
<!ELEMENT ville (#PCDATA)>
<!ELEMENT pays (#PCDATA)>
<!ELEMENT longitude (#PCDATA)>
<!ELEMENT latitude (#PCDATA)>
<!ELEMENT datedebut (jour, mois, annee)>
<!ELEMENT datefin (jour, mois, annee)>
<!ELEMENT annee (#PCDATA)>
<!ELEMENT mois (#PCDATA)>
<!ELEMENT jour (#PCDATA)>
<!ELEMENT ouvert_bool (#PCDATA)>
<!ELEMENT heure_ouv_matin (heure, minute)>
<!ELEMENT heure_ferm_matin (heure, minute)>
<!ELEMENT heure_ouv_PM (heure, minute)>
<!ELEMENT heure_ferm_PM (heure, minute)>
<!ELEMENT heure (#PCDATA)>
<!ELEMENT minute (#PCDATA)>
<!ELEMENT libelle (#PCDATA)>
<!ELEMENT nombre (#PCDATA)>
<!ELEMENT libelle_age (#PCDATA)>
<!ELEMENT valeur (#PCDATA)>
<!ELEMENT telephone (indic_tel, num_tel)>
<!ELEMENT fax (indic_tel, num_tel)>

```

```

<!ELEMENT indic_tel (#PCDATA)>
<!ELEMENT num_tel (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT url (#PCDATA)>
<!ELEMENT menu_enfant_bool (#PCDATA)>
<!ATTLIST menu_enfant_bool prix_menu_enfant CDATA #IMPLIED>

```

Lors de l'écriture et de certains tests de la DTD et de sa structure, nous avons procédé à quelques changements minimes sur le schéma ERA que vous trouverez à la figure 5.3 page suivante.

À présent, nous allons définir la sémantique puis la syntaxe de chaque élément du fichier XML afin de permettre d'avoir une vision claire des fiches XML. Nous trouvons en annexe du mémoire, le code source de 3 fichiers d'exemple XML ; un fichier pour chaque catégorie d'objet : hébergement, restaurant, activité.

### 5.5.2 Sémantique des éléments de la DTD

Afin de définir la sémantique des différents éléments des fiches touristiques, nous allons repartir du schéma ERA de la figure 5.3 page ci-contre afin de l'expliquer et d'expliquer/justifier les choix que nous avons fait. Après il faudra rechercher le meilleur moyen de définir sa structure physique et surtout les types utilisés car la DTD ne permet pas de faire cela (il est juste possible de manipuler des #PCDATA). Il faudra alors peut-être se tourner du côté des XML Schema et voir si il nous est possible de redéfinir ses propres types comme on peut le faire en programmation orienté objet grâce à des mécanismes de surcharges et d'héritages. Sinon il faudra voir le meilleur compromis possible afin d'avoir quelque chose de bien défini et éviter de devoir uniquement travailler avec des chaînes de caractères dont on aurait simplement défini une certaine syntaxe pour rendre son traitement possible.

L'entité **objet** regroupe l'ensemble des informations communes à tous les types de fiches touristiques (hébergement, restaurant et activité). Ainsi, on retrouve le nom (le nom du restaurant, de l'hôtel, du camping, du musée, etc...), son type (hébergement, restaurant, activité), le sous-type qui dépend du type précédemment indiqué (nous donnerons le détail par la suite).

D'un point de vue purement orienté base de données, ces attributs (tout du moins une partie) sont redondants car il est possible de dériver cette information (type d'objet que nous manipulons) de l'IS-A. Néanmoins, d'un point de vue purement pratique et pour une meilleure compréhension de tous (surtout pour un public non initié au domaine des bases de données) mais également du fait que nous travaillons avec des fichiers XML et non des bases de données, il nous est nécessaire d'indiquer un tel attribut. Il faut donc indiquer que notre schéma ERA n'est pas totalement normalisé mais c'est un choix que nous avons fait.

Revenons au détail du champ sous-type. Voici donc les valeurs possibles de ce champ en fonction de la valeur du champ **type** :

- hébergement → hôtel | camping | mobilhome | chambre d'hôte | gîte | location
- restaurant → gastronomique | brasserie | snack/fast-food
- activité → sportive | culturelle | manuelle

Deux mots sur la signification de certains termes. Tout d'abord **mobilhome** concerne les types de location que l'on trouve dans les campings, autre qu'un emplacement pour une caravane ou une tente mais des caravanes résidentielles, des chalets, des logements mobiles par exemple. **Location** concerne la location d'un appartement, d'une maison, d'une villa, etc... pour une ou plusieurs semaines.

Concernant le type restaurant, les termes employés nous paraissent suffisamment clairs et compréhensibles par tous de la même manière. Concernant les activités, **sportive** indique par exemple une randonnée VTT, une promenade à cheval, n'importe quel sport, etc..., **culturelle** regroupe les musées, les expositions, les visites historiques, etc... et enfin, **manuelle** indique plus des stages ou séances d'initiation à la peinture, à la poterie, etc... ou toute autre activité manuelle.

Le groupe d'attribut **coordonnées** regroupe toutes les informations sur la manière de pouvoir contacter



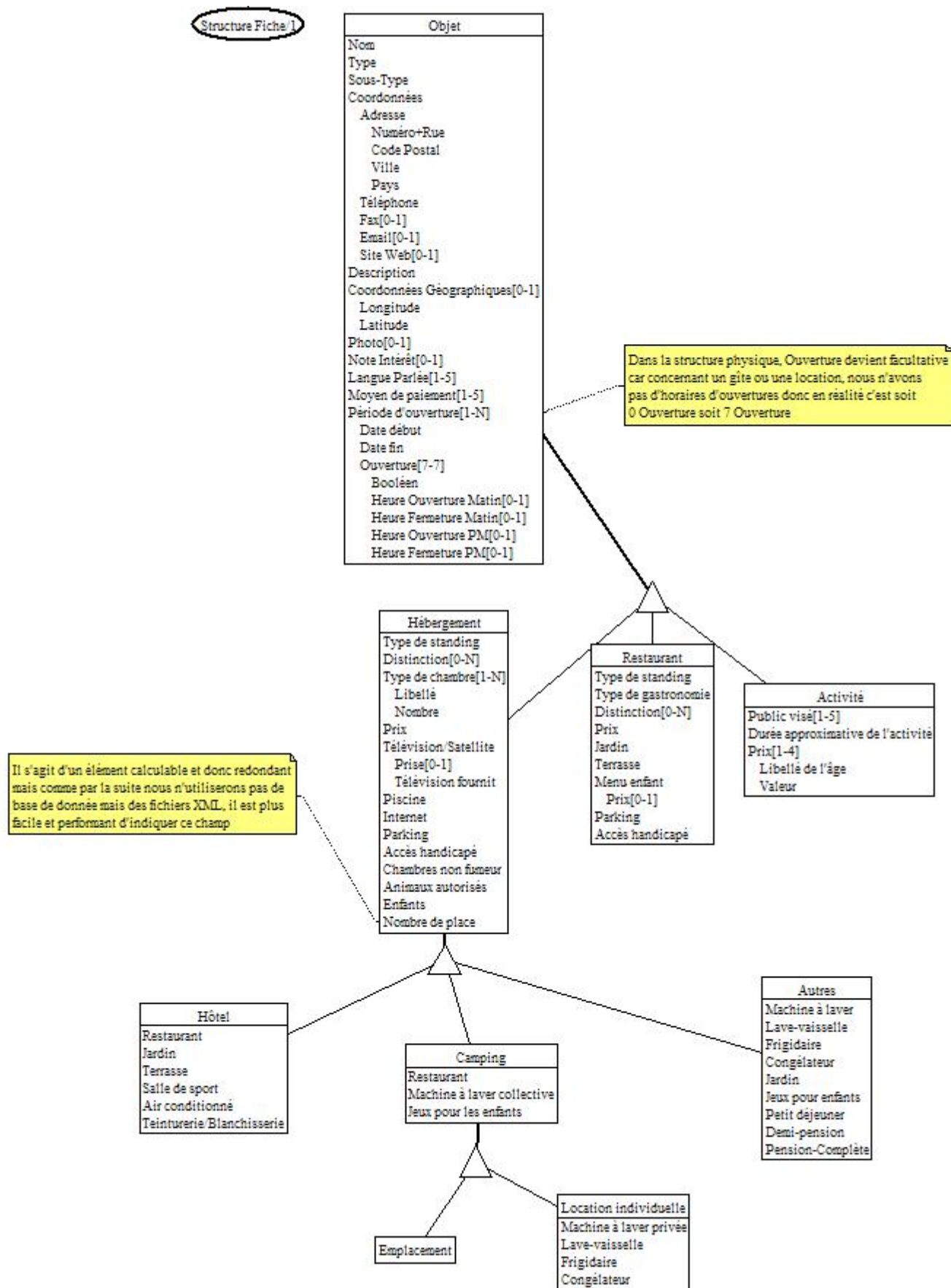


Figure 5.3 – Version finale du schéma ERA de la structure des fiches

l'hébergeur, le restaurateur ou bien le responsable de l'activité et obtenir des renseignements. On trouve donc l'**adresse**, le **numéro de téléphone**, le **fax**, l'**email** ainsi que l'**adresse du site web**. Il faudra définir ou utiliser un type de donnée spécifiques pour ces éléments.

Ensuite on retrouve la **description** de l'objet qui est un texte assez court. On a également les **coordonnées géographiques** composées de la **longitude** et de la **latitude** afin de pouvoir par la suite situer le lieu sur une carte électronique de l'IGN ou de Google par exemple ou bien proposer un itinéraire pour se rendre sur le lieu, etc... Il faudra également définir un type particulier pour cet attribut. Le champ **photo** est en fait un lien (ou url) vers un fichier image pour donner un aperçu à l'utilisateur du lieu.

Le champ **note d'intérêt** permet de donner une appréciation personnelle de l'objet touristique comme on le retrouve fréquemment dans les livres touristiques avec souvent des étoiles (de 0 à 3 étoiles). On pourrait donc utiliser un entier qui serait compris entre 0 et une certaine valeur pour juger l'objet et donner une idée aux futurs touristes, même si il est évident qu'une note est quelque chose d'assez subjectif. Le champ multivalué **langues parlées** permet de renseigner les visiteurs des langues parlées et comprises par les propriétaires et les employés de l'objet. Nous avons limité à 5 le nombre de langues possibles car dans la réalité, il est rare de trouver un établissement où l'on parle plus de 5. Il s'agit juste d'un choix technique qui est facilement modifiable. De la même manière que pour les langues, le champ **moyen de paiement** permet d'indiquer aux consommateurs les moyens acceptés par l'établissement (Visa, Mastercard, chèque, ...). Pour les mêmes raisons, nous avons limité à 5 le nombre de moyens possibles mais on peut tout aussi facilement que pour les langues, le modifier.

Enfin le dernier groupe d'attributs de cette entité est celui de la ou des **période(s) d'ouverture** de l'établissement. Il est composé tout d'abord d'une **date de début** et d'une **date de fin** ainsi que d'un **groupe** de 7 éléments, **ouverture**. Il permet ainsi d'indiquer pour chaque jour de la semaine si l'établissement est ouvert et si oui, quelles sont les heures d'ouverture et de fermeture du matin et de l'après-midi. Ainsi, si le champ booléen est à false, les heures ne doivent pas exister et dans le contraire au moins une heure d'ouverture et une heure de fermeture est obligatoire.

Intéressons nous maintenant à l'entité **activité**. Le premier attribut, **public visé**, permet d'indiquer une liste de personnes susceptibles de pouvoir apprécier l'activité. Il s'agira d'une chaîne de caractères pouvant prendre par exemple les valeurs suivantes :

- enfant
- adolescent
- adulte
- étudiant
- ...

Le deuxième attribut, **durée approximative de l'activité**, permet d'indiquer à l'utilisateur combien de temps environ il doit prévoir pour faire la visite, la promenade, etc... et ainsi planifier sa journée. Cette information sera de type time sûrement sous le format HH :MM. Enfin, l'attribut prix que nous ne détaillons plus car il a déjà été abordé dans la section précédente.

Concernant les restaurants, il y a un attribut **type de standing** qui permet d'informer le client sur le type d'établissement et son confort (on retrouve cette information sur le site internet de Michelin avec les critères suivants : grand luxe, grand confort, très confortable, de bon confort, assez confortable, Pubs/Tapas. Nous pensons ces critères satisfaisants dans cet état). De toute manière, il sera très facile d'ajouter/supprimer des valeurs à la liste possible de valeurs. L'attribut **type de gastronomie** permet d'indiquer le type de plat servi dans le restaurant. Il faudra voir si il est utile de proposer une liste de valeurs possibles ou bien si on utilise simplement une chaîne de caractères car il existe de nombreuses cuisines à travers le monde et cela serait peut-être trop restrictif de proposer une liste de valeurs. Le champ multivalué facultatif **distinction** permet d'indiquer si l'établissement a reçu des distinctions telles que les étoiles du guide Michelin, la note du GaultMillau, etc. afin de mettre en valeur le talent du chef du restaurant. Le champ **prix** sera d'un type intervalle de prix afin de donner le prix minimum et le prix

maximum des menus du restaurant car sur une fiche touristique, il est évident qu'il est impossible d'indiquer l'ensemble des prix des menus et leur composition. Il faudra donc définir deux types de donnée : le type intervalle ainsi qu'un type prix basé sur le type monétaire des bases de données. Ensuite, nous avons une liste de booléens afin de pouvoir donner un certain nombre de caractéristiques de l'établissement et permettre un affichage libre sur la page internet. Ainsi nous avons un champ **jardin** (pour dire si il y a un jardin au restaurant), un champ **terrasse** (afin d'indiquer si il est possible de manger en extérieur l'été quand il fait beau), un champ **menu enfant** pour indiquer si le restaurant propose un tel menu (et un champ facultatif **prix** si le booléen est égale à true et que nous possédons l'information), un champ **parking** (pour indiquer si l'établissement dispose d'un parking privé pour pouvoir se stationner facilement) et enfin l'attribut **accès handicapé** (pour indiquer si le restaurant possède l'équipement et les installations nécessaires pour accueillir les personnes atteintes d'un handicap physique). L'ensemble de ces booléens nous permettrons ainsi, lors de notre réflexion sur l'affichage des informations des fiches, de pouvoir exploiter le true ou le false de la manière dont il nous plaira et aussi de dégager des valeurs discriminantes par rapport à l'ensemble des résultats de notre recherche.

Il reste maintenant à nous intéresser aux logement, type d'objet plus complexe car possédant différents sous-types comme hôtel, camping et autres.

Commençons d'abord avec l'entité **hébergement** et ses attributs qui sont communs à tous les sous-types. Comme pour le restaurant, il existe un attribut **type de standing** avec la même sémantique donc nous ne revenons pas dessus. Idem pour l'attribut multivalué facultatif **distinction**. L'attribut multivalué **type de chambre** permet de lister les différentes configurations possibles pour les chambres (chambre 1 lit 2 personnes, chambre 2 lits 1 personne, avec baignoire, avec douche, suite, etc...) ; cet attribut est composé d'un **libellé** qui indique le type de la chambre sous forme d'une chaîne de caractères et du **nombre** de ce type de chambre que possède l'hébergement afin de pouvoir afficher une répartition en fonction du type de chambre. Nous avons également un attribut **prix** qui comme pour les restaurants sera sous la forme d'un intervalle. Nous avons également un groupe d'attribut **télévision** composé de 2 attributs : **prise** qui est facultatif et **télévision fournit**. Ce sont deux booléens qui permettent d'indiquer si l'hébergement possède une prise de télévision ainsi que si une télévision est fournit dans la location. En effet, parfois dans des mobilhomes ou bien des locations, il y a simplement la prise mais la télévision en elle-même n'est pas fournit. Ensuite, nous avons une liste de booléens qui permettent de lister différents éléments présents ou non dans l'hébergement : **piscine**, **connexion internet**, **parking**, **accès handicapé**, **chambres non fumeur**, **animaux autorisés**, **enfants acceptés**. Pour ces attributs, il n'y a pas besoin d'explications supplémentaires. Il reste enfin le champ **nombre de place** qui indique le nombre de places total disponibles dans l'hébergement afin qu'immédiatement, les personnes puissent savoir si l'hébergement est suffisamment grand pour eux ou pas. Ce champ, du point de vue ERA et des bases de données, est un champ calculable et donc redondant. Il ne faudrait pas le stocker car il suffit de faire une boucle sur les types de chambre et d'additionner le nombre de chaque chambre mais étant donné que nous sommes au niveau d'un fichier XML, le calcul n'est pas aussi simple et efficace que dans une base de donnée et donc il est plus aisé d'ajouter ce champ numérique.

Maintenant intéressons-nous à l'entité **hôtel**. Il possède une liste d'attributs booléens supplémentaires par rapport à l'entité hébergement afin d'indiquer des caractéristiques supplémentaires que l'on pourrait trouver dans un hôtel. Ainsi, nous retrouvons **restaurant** pour indiquer si l'hôtel possède un restaurant rattaché où les clients peuvent manger, les attributs **jardin**, **terrasse** qui ont le même sens que pour le restaurant, l'attribut **salle de sport** pour indiquer si il y a une salle de sport où l'on peut faire du sport dans l'hôtel, l'**air conditionné** dans les chambre et enfin **teinturerie/blanchisserie** où les clients pourraient déposer leur linge sale à nettoyer et repasser.

Nous avons ensuite l'entité **Camping** qui possède deux sous-type : **emplacement** et **location individuelle**. **Emplacement** correspond à la location d'un emplacement pour une caravane, un camping-car ou une tente tandis que location individuelle correspond à la location de chalet, mobilhome, roulotte, etc... On retrouve les booléens suivants : **restaurant**, **machine à laver collective** et **jeux pour les enfants** ; restaurant indiquant si il existe un restaurant ou un snack à l'intérieur du camping où il est

possible de manger, machine à laver collective indiquant si il existe un lavoir automatique ou quelque chose dans le même style disponible à tous les clients du camping mais qui peut être éventuellement payant évidemment. Pour l'entité **emplacement**, nous avons indiqué également la **superficie** en m<sup>2</sup>. Pour les **locations individuelles**, nous avons indiqué les booléens suivants : **machine à laver privée** à l'intérieur de la location qui est accessible uniquement aux clients de ce mobilhome, chalet ou autre... les booléens **lave-vaisselle**, **fridaire** et **congélateur** également.

Enfin, il reste l'entité **autres** qui regroupe les locations d'appartement, maison, etc... appartenant à un particulier, ainsi que les gîtes, les maisons d'hôtes. Nous avons les booléens suivants : **machine à laver**, **lave-vaisselle**, **fridaire**, **congélateur**, **jardin** qui possèdent le même sens que précédemment et également **jeux pour les enfants** pour indiquer qu'il existe dans le jardin ou dans la maison des équipements pour les enfants (balançoire, toboggan, etc...). Un peu plus particulier aux maisons d'hôte ou les gîtes, des **booléens petit déjeuner**, **demi-pension**, **pension-complète** afin d'indiquer si ces services sont offerts ou pas par le propriétaire des lieux.

Maintenant que le travail de sémantique est terminé, il va falloir réfléchir à la manière de stocker les informations dans le fichier XML (quel type de donnée pour chaque attribut) et d'en faire un XML Schema qui permet de travailler plus en détail sur les types de donnée que ne le permet la DTD.

### Précisions sur les XML Schema

Afin de clarifier certains éléments concernant les XML Schema par rapport à la DTD, nous avons organisé une réunion d'information avec Monsieur Thierry Despeyroux<sup>3</sup> qui travaille sur XML depuis longtemps. Nous espérons que suite à cette réunion, nous aurons les idées claires et une bonne vision des différences entre les deux et pourquoi pas certains conseils concernant l'utilisation de chacun.

Comme nous l'avons déjà fait remarqué auparavant, il a été confirmé par Thierry qu'il n'existe pas de types dans les DTD et que cela existait dans les XML schema (on trouve la plupart des types de variables qui existent dans tous les langages de programmation ainsi que la possibilité de créer des types complexes équivalent aux records en Pascal par exemple) mais qu'il n'a jamais eu vraiment l'occasion de les utiliser. Afin de pouvoir mieux nous conseiller et orienter dans notre choix, il nous a posé alors les questions de ce que nous avons besoin réellement à notre niveau : « Avons-nous besoin de faire une vérification stricte des types ou bien pouvons-nous considérer que nos fichiers XML sont corrects et simplement les utiliser pour les informations contenues à l'intérieur ? Quelle est la méthode de création des fiches (saisie manuelle, récupération de l'information depuis une base de donnée ou bien d'Internet, etc...) et la source est-elle fiable ? » Après réflexion avec Yves Lechevallier, nous avons pris l'hypothèse que les fichiers XML étaient correctes et que les « types » étaient respectés (par exemple pour les heures, nous émettons l'hypothèse qu'elle est correcte et que nous ne pouvons pas avoir 25 :67 par exemple et idem pour les dates ou bien si il s'agit d'une information numérique, nous ne pouvons pas avoir de lettres ou ce genre d'erreurs qui pourraient compromettre la validité des fiches et leur affichage par la suite) car en effet, notre recherche se focalise sur la représentation graphique de l'information et pas sur la création/extraction de contenu et sa validation. Finalement, les DTD sont suffisantes pour cela car les XML schema alourdissent leur gestion et que cela ne se justifie pas dans notre cas.

Nous allons donc définir maintenant les formats de données que l'on attend pour chaque élément et pouvoir commencer ensuite la réflexion sur la représentation graphique.

### 5.5.3 Syntaxe des éléments de la DTD

Après avoir explicité la sémantique, intéressons-nous maintenant à la syntaxe des différents éléments. Seuls les éléments déclarés en #PCDATA dans la DTD sont concernés car les autres sont des éléments composés d'autres éléments et donc, seuls les éléments finaux ont une syntaxe précise. Par soucis de simplicité, nous avons décidé de reprendre les éléments dans l'ordre de la DTD.

---

<sup>3</sup>Chercheur à l'INRIA de Rocquencourt

- rest\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- jardin\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- terrasse\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- salle\_sport\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- clim\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- blanchisserie\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- superficie : il s'agit d'un numérique strictement positif exprimé en m<sup>2</sup> mais nous ne stockons pas le m<sup>2</sup> dans le fichier XML. Ainsi par exemple 25 m<sup>2</sup> sera simplement 25 dans le fichier.
- machine\_laver\_privée\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- lave\_vaisselle\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- frigo\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- congelateur\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- machine\_laver\_collective\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- jeux\_enfants\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- petit\_dej\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- demi\_pension\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- pension\_compl\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- type\_standing : il s'agit d'une chaîne de caractères avec une liste de termes prédéfinis qu'il reste à définir et qui pourra être complétée par la suite car le mécanisme est simple. Cette liste sera sûrement basée sur ce que fait Michelin dans ses différents guides
- type\_gastronomie : il s'agit d'une chaîne de caractères avec une liste de termes prédéfinis qui sont française, italienne, méditerranéenne, chinoise, japonaise, ... cette liste n'est pas fixe et peut être

complétée à tout moment

- distinction : il s'agit d'une chaîne de caractères sans liste de termes prédéfinis car le sujet est vraiment vague (dépend des guides, des départements, ...) et donc il ne paraissait pas utile d'y inclure une liste de termes
- prix : il s'agit d'un intervalle de deux valeurs numériques décimale à 2 chiffres après la virgule. Nous ne stockerons pas le symbole monétaire de l'euro dans le fichier XML mais simplement la valeur. La valeur de gauche doit évidemment être inférieure à celle de droite. Dans le cas où il s'agit d'un prix fixe et non d'un intervalle, la valeur gauche et la valeur droite sont égales. Pour définir la notion de prix jusqu'à x€, on noterait [0.00;x] et pour la notation à partir de x€, on noterait [x;9999999.99] sinon pour un intervalle de prix, on aurait [x;y]. Le string est donc écrit de la manière suivante : crochet ouvrant, la première valeur, un point-virgule, la deuxième valeur, crochet fermant, c'est-à-dire **[x;y]**. On utilise le point comme symbole de séparation pour la partie décimale d'une valeur
- prise\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- television\_fournit\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- piscine\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- internet\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- parking\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- acces\_hand\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- non\_fumeur\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- animaux\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- enfants\_bool : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur true ou false (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- nbr\_place : il s'agit d'un numérique entier strictement positif
- public\_vise : il s'agit d'une chaîne de caractères avec les termes prédéfinis suivants : enfant, adolescent, adulte, étudiant mais qui pourra être complétée afin d'ajouter une information plus précise sur l'âge comme enfant de moins de 6 ans, etc...
- nom : il s'agit d'une chaîne de caractères sans aucune valeur prédéfinie
- description : il s'agit d'une chaîne de caractères d'une centaine de mots
- photo : il s'agit de l'url vers la photo de l'objet et elle peut être de type absolue (par exemple [www.monsite.com/photo/1.jpg](http://www.monsite.com/photo/1.jpg)) ou bien de type relatif ([../photo/1.jpg](#))
- noteinteret : il s'agit d'un numérique compris entre 0 et 5 par exemple (voir 10 ou 20 mais après cela risque de faire trop et les personnes ne sauront pas faire trop la différence entre un 15 et un 16 par exemple alors que c'est plus facile d'imaginer entre 3 et 4) qui pourra ensuite lors de la représentation graphique changer en une autre notation

- `langueparlee` : il s'agit d'une chaîne de caractères parmi une liste de valeurs prédéfinies des langues principales parlées
- `paiement` : il s'agit d'une chaîne de caractères parmi une liste des valeurs suivantes : espèce, chèque, CB, Visa, Mastercard, American Express qui peut être complétée si nécessaire par la suite. On pourra par la suite, au moment de la représentation graphique, utiliser cette information différemment
- `numero` : il s'agit d'un numérique entier strictement positif
- `rue` : il s'agit d'une chaîne de caractères
- `code_postal` : il s'agit d'un numérique positif sur 5 positions pour la France (+/- de chiffres pour les autres pays et parfois même alphanumérique)
- `ville` : il s'agit d'une chaîne de caractères
- `pays` : il s'agit d'une chaîne de caractères
- `longitude` : il s'agit d'un numérique décimal compris entre 0 et 180 avec jusqu'à 5 ou 6 chiffres après le symbole de séparation. Parfois, on pourrait avoir un alphanumérique pour ajouter en plus de la valeur décimal, une lettre pour indiquer la direction (E pour Est et O pour Ouest) mais cela dépend de la notation utilisée. Il faudra donc s'informer auprès de Google Maps ou de l'IGN pour voir quelle est la notation gérée par leur API
- `latitude` : il s'agit d'un numérique décimal compris entre 0 et 90 avec jusqu'à 5 ou 6 chiffres après le symbole de séparation. Parfois, on pourrait avoir un alphanumérique pour ajouter en plus de la valeur décimal, une lettre pour indiquer la direction (N pour Nord et S pour Sud) mais cela dépend de la notation utilisée. Il faudra donc s'informer auprès de Google Maps ou de l'IGN pour voir quelle est la notation gérée par leur API
- `annee` : il s'agit d'un numérique strictement positif sur 4 positions
- `mois` : il s'agit d'un entier compris entre 1 et 12
- `jour` : il s'agit d'un entier compris entre 1 et 31
- `ouvert_bool` : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur `true` ou `false` (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- `heure` : il s'agit d'un entier compris entre 0 et 23
- `minute` : il s'agit d'un entier compris entre 0 et 59
- `libelle_age` : il s'agit d'une chaîne de caractères pouvant prendre une valeur parmi la liste suivante : enfant, adulte, groupe, tarif réduit
- `valeur` : il s'agit d'une valeur numérique décimale à 2 chiffres après la virgule. Nous ne stockerons pas le symbole monétaire de l'euro dans le fichier XML mais simplement la valeur. Ainsi un prix de par exemple 23.99€ sera enregistré ainsi 23.99 avec le point comme symbole de séparation
- `indic_tel` : il s'agit en principe d'un entier positif mais maintenant on a tendance à souvent remplacer le 00 par + depuis l'explosion des téléphones portables donc cela serait plutôt une chaîne d'alphanumérique
- `num_tel` : il s'agit d'une valeur numérique positive
- `email` : il s'agit d'une chaîne alphanumérique comportant obligatoire un symbole arobase @ et d'au moins un point suivant l'expression régulière suivante :  
`/^[a-zA-Z][a-zA-Z0-9-_.]+@([-a-zA-Z]+\.)+[a-zA-Z]{2,6}/`
- `url` : il s'agit d'une chaîne de caractères alphanumérique vérifiant l'expression régulière suivant :  
`^http(s)?://[-[:alnum:]]+\. [-[:alnum:]]+\. [a-zA-Z]{2,4}`
- `menu_enfant_bool` : il s'agit d'un booléen qui sera représenté par un string pouvant prendre la valeur `true` ou `false` (on aurait pu également choisir 1 ou 0 mais cela permet plus de clarté dans le code du fichier XSL ensuite)
- `prix_menu_enfant` : il s'agit d'une valeur numérique décimale à 2 chiffres après la virgule. Nous ne stockerons pas le symbole monétaire de l'euro dans le fichier XML mais simplement la valeur. Ainsi un prix de par exemple 23.99€ sera enregistré ainsi 23.99 avec le point comme symbole de

séparation

Maintenant que nous avons défini et structuré l'information pour les futures fiches touristiques et avant de nous pencher sur la visualisation de ces dernières, nous allons aborder les problèmes d'IHM tels que l'accessibilité et l'ergonomie dans le web en général et puis dans le domaine de la mobilité en particulier.



Depuis des années maintenant, de nombreux chercheurs de diverses formations ont travaillé dans le domaine de l'accessibilité et de l'ergonomie. En effet, nous retrouvons des ergonomes possédant une formation en science cognitive et psychologique mais également des informaticiens.

Le but de ces recherches est principalement de réfléchir au moyen de définir des règles et recommandations pour développer des interfaces utilisables par tous, y compris les personnes souffrant d'handicaps. En se basant sur ces différentes recherches, le W3C, au niveau international, a travaillé à l'adaptation de ces règles au domaine du web et depuis peu, également pour le web sur périphérique mobile. Au niveau francophone, d'autres projets ont vu le jour comme le projet Anysurfer<sup>1</sup> (anciennement Blind Surfer) au niveau belge, AccessiWeb<sup>2</sup> en France, Destine aux facultés universitaires Notre-Dame de la Paix de Namur sur lesquels nous reviendrons plus loin.

Maintenant que nous avons une structure sémantique complète et que nous avons défini avec précision le contenu que nous souhaitons afficher sur les fiches touristiques, nous allons réfléchir à la manière de mettre ces informations en page (le layout). Nous allons donc nous appuyer sur les documents du W3C [53] et [52] ainsi que sur le livre [29] (bien que traitant principalement des IHM de logiciels, plusieurs chapitres sont néanmoins consacrés aux sites internet).

Ainsi, dans un premier temps, nous allons exposer les grandes lignes des différentes sources d'informations sur le sujet afin ensuite de dresser un premier layout des fiches. Ensuite, une fois la mise en page effectuée, nous développerons différentes solutions de visualisations pour chaque type d'information contenu dans les fiches ainsi que pour la page de recherche d'une information sur le site. Enfin, nous conclurons par une étude comparant les requêtes écrites par rapport à celles parlées dont les résultats nous seront utiles pour nos scénarios d'usages.

## 6.1 Recommandations du W3C concernant l'accessibilité

Les recommandations du W3C sont organisées suivant quatre principes fondamentaux [53] :

1. le contenu doit être perceptible
2. les composants de l'interface dans le contenu doivent être fonctionnels
3. le contenu et les contrôles doivent être compréhensibles

---

<sup>1</sup>[www.anysurfer.be](http://www.anysurfer.be)

<sup>2</sup>[www.accessiweb.org/fr/guide\\_accessiweb/index.html](http://www.accessiweb.org/fr/guide_accessiweb/index.html)

4. le contenu doit être suffisamment robuste pour fonctionner avec les user agents actuels et futurs

Ainsi, de ces 4 principes fondamentaux, des critères vont suivre afin de pouvoir les faire respecter sur internet. Pour chacun d'entre eux, il est indiqué les outils technologiques à mettre en place pour y parvenir. Le W3C a mis en place 3 niveaux de succès de ces critères.

Le niveau 1 de succès permet d'acquérir un niveau minimal d'accessibilité et peut être raisonnablement appliqué à tous les contenus web. Le niveau 2 permet d'acquérir un niveau avancé d'accessibilité et peut être également appliqué raisonnablement à tous les contenus. Enfin, le niveau 3 d'acquérir des avancées d'accessibilité supplémentaires et ne peut pas être appliqué à tous les contenus web. C'est pour cela que pour être en conformité avec les critères du W3C il n'est pas nécessaire de répondre à l'ensemble des critères du niveau 3.

Afin de pouvoir classer les sites en fonction du respect des recommandations d'accessibilité, des niveaux de conformités ont été mis en place par le W3C. Ils sont au nombre de 3 :

1. niveau A : tous les critères du niveau 1 des critères de succès sont vérifiés
2. niveau AA : tous les critères du niveau 1 et 2 des critères de succès sont vérifiés
3. niveau AAA : tous les critères du niveau 1, 2 et au moins la moitié des critères du niveau 3 sont vérifiés

Évidemment, lors de l'absence d'un type de contenu sur la page testée à évaluer, le critère est considéré comme vérifié.

Sans rentrer dans le détail de tous les critères à vérifier et la manière de réussir l'évaluation des quatre principes fondamentaux de l'accessibilité, nous allons revenir sur les attentes du W3C concernant chacun de ces principes sans expliquer comment parvenir à combler ces attentes.

- Le principe 1 consiste à fournir un texte alternatif à tout contenu non-textuel, de fournir des alternatives synchronisées pour le contenu multimédia. On doit également s'assurer que l'information l'information et la structure de la page peuvent être séparées de la présentation. Enfin, on doit pouvoir facilement distinguer l'information de premier plan et de celle d'arrière plan.
- Le principe 2 consiste à rendre tous les fonctionnalités utilisables avec un clavier, permettre à l'utilisateur de contrôler les limites de temps pour la lecture du contenu ou l'interaction avec ce dernier. On doit également permettre aux utilisateurs d'éviter un contenu qui pourrait occasionner une crise (d'épilepsie par exemple) due à la photo-sensibilité. On doit également fournir des mécanismes pour aider les utilisateurs à trouver le contenu, l'orienter vers ce qu'il cherche et l'aider à naviguer à l'intérieur de ce contenu. On doit également l'aider à éviter de faire des erreurs ou bien de lui permettre de les corriger facilement.
- Le principe 3 consiste à faire en sorte que le texte soit lisible et compréhensible, rendre le placement et les fonctionnalités du contenu prévisibles.
- Le principe 4 consiste à offrir une compatibilité du contenu avec les user agents actuels et futures et s'assurer que ce contenu soit accessible ou bien fournir une alternative accessible de ce contenu.

Tous ces critères sont là afin de pouvoir rendre accessible à tous (même les personnes souffrant de certains handicaps) le contenu du web. En effet, ces personnes utilisent des user agents classiques (comme Internet Explorer, Firefox, Windows Media Player, etc.) mais sur lesquels ils ont « greffé » des plugins afin d'enrichir les fonctionnalités de bases par d'autres spécialisées dans des domaines comme dans la synthèse vocale, la présentation simplifiée, etc. (comme Lynx par exemple) et donc ces logiciels doivent pouvoir traiter correctement l'information de la page afin de pouvoir la transcrire différemment à l'utilisateur (lecture de la page par exemple).

Dans le document des recommandations, on trouve également des explications sur la manière de réussir le test des différents critères ainsi qu'une grille d'analyse d'un site afin d'indiquer si le critère est vérifié ou non et d'éventuels commentaires pour chacun d'eux. Nous essayerons donc, dans notre recherche de mise en page pour les fiches touristiques de vérifier ces recommandations mais le document émis également par le W3C concernant les bonnes pratiques pour le web mobile nous sera plus utile dans le cadre de nos recherches. Dans le livre [29], un chapitre est consacré aux interfaces web : on trouve ainsi ce qui les caractérisent et ce qui les différencient par rapport aux interfaces graphiques utilisateurs (GUI en

anglais) traditionnelles. Nous considérons que ces informations sont connues par le plus grand nombre et nous n'avons donc pas jugé utile de les traiter en détails ici. Si vous souhaitez plus d'informations à ce sujet, vous pouvez toujours consulter [29].

## 6.2 Évaluation automatique de l'ergonomie des site web

Nous avons vu que des organismes travaillent sur l'élaboration de recommandations concernant le développement de sites web afin qu'ils soient faciles à parcourir et utilisable par tous, mêmes les utilisateurs handicapés. Le W3C a mis en place un système de validation des balises HTML, XHTML et CSS mais des projets novateurs sont en cours de développement afin d'évaluer de manière automatique et manuelle (pour certains éléments qui ne peuvent pas être évalués autrement) l'ergonomie des sites web sous différents aspects. On peut citer pour cela le projet DESTINE mené aux facultés universitaires de Namur (FUNDP<sup>3</sup>) en collaboration avec l'UCL<sup>4</sup>.

Sans rentrer dans les détails, nous pouvons reprendre le résumé de [41] afin d'expliquer les grandes lignes du projet : *L'Soutil DESTINE est conçu en vue de l'évaluation de l'ergonomie des sites web. L'évaluation se fait de manière automatique pour certaines règles, et manuelle pour les règles non automatisables. DESTINE couvre l'entièreté de la tâche d'évaluation, en partant de la spécification du contexte jusqu'à la finalisation du rapport d'évaluation, et sa communication.*

Nous allons maintenant aborder, avant d'étudier les différentes recommandations spécifiques au web mobile, aux problèmes d'impression des pages web.

## 6.3 Page imprimée vs page web

Le dernier élément important à prendre en compte lors d'application web, c'est l'impression des pages. En effet, sur des applications bureautiques, il n'existe pas d'impression standard des données. Le développeur doit donc programmer des rapports pour imprimer les données saisies et stockées par exemple dans des bases de données. La mise en page et la présentation est donc réfléchie et identique quelque soit l'imprimante, l'ordinateur utilisé, etc. Dans le domaine du web, les choses sont différentes. En effet, tous les navigateurs web proposent d'imprimer la page en cours mais le résultat obtenu est rarement satisfaisant. Nous avons des pages blanches, le texte est tronqué, la présentation pas agréable à lire, etc. Le développeur web, au départ n'a pas la main mise sur le résultat final sorti de l'imprimante. À chaque nouvelle version des navigateurs web, les éditeurs essaient d'améliorer l'impression des pages mais il reste encore pas mal de travail à faire. En effet, de plus en plus, on trouve des animations Flash sur les pages, des vidéos, ... et ce n'est pas possible de les imprimer ou bien alors de manière fixe, avec une vignette image.

C'est pour cela que l'on voit apparaître de plus en plus, sur les sites commerciaux, des « versions imprimables » des pages. On a par exemple une icône représentant une imprimante en bas de la page qui permet d'afficher une **version optimisée** de la page où le développeur a généralement supprimé les publicités, certaines images, les animations ou vidéos, etc. tout ce que nous n'avons pas besoin sur le papier. Parfois, la mise en page est également modifiée afin de mieux mettre le contenu important en évidence et d'une manière plus agréable pour la lecture ensuite. L'impression des pages est donc un facteur important à prendre en compte, surtout de nos jours où les mises en pages des sites sont de plus en plus complexes. Les feuilles de styles (CSS) est une première solution pour y parvenir mais il existe évidemment d'autres techniques encore plus efficaces mais nécessitant plus de travail pour y parvenir. On pourrait par exemple penser à une analyse complète de la structure HTML de la page afin d'en dégager un arbre syntaxique et remplacer certains types de nœuds en d'autres plus adaptés pour l'impression.

Néanmoins, même en voulant vraiment développer une solution pour l'impression des pages web, le

---

<sup>3</sup>[www.fundp.ac.be](http://www.fundp.ac.be)

<sup>4</sup>[www.uclouvain.be](http://www.uclouvain.be)

développeur ne peut pas être garanti du résultat final car comme pour le rendu et l’affichage d’une page, l’impression variera d’un navigateur web à un autre et donc nous n’avons aucune garantie du résultat et il est impossible d’uniformiser à chaque ordinateur le résultat pour une même page donnée.

Sinon, il est possible d’offrir la possibilité de générer un document PDF par exemple de la page en cours afin de pouvoir l’ouvrir et l’imprimer facilement sur n’importe quel ordinateur. Il existe une version pour Pocket PC d’Acrobat Reader donc il est également possible de l’ouvrir sur son PDA.

Maintenant que nous avons abordé les caractéristiques propres aux interfaces web par rapport aux GUI ainsi que sur les règles d’accessibilité, il est temps d’appliquer tout cela aux périphériques mobiles, sujet qui nous intéresse dans ce travail. Nous allons donc dans un premier temps, étudier les recommandations données par le W3C concernant le web sur les périphériques mobiles et par la suite, nous réfléchirons à la solution que nous souhaitons proposer pour la mise en page des fiches touristiques.

## 6.4 Recommandations du W3C pour le web mobile

Dans le document [53], le W3C nous donne un certain nombre de recommandations concernant la manière de développer un site web destiné aux périphériques mobiles. Comme pour les recommandations au sujet de l’accessibilité, nous n’allons reprendre ici que les points importants que nous pourrions avoir besoin pour nos fiches touristiques.

### 6.4.1 Navigation et liens

Tout d’abord, dans ce document, le W3C préconise pour l’ensemble du développement web (afin de ne pas devoir développer plusieurs sites différents : un pour chaque type de périphériques) d’avoir « un seul web » ("One Web" en anglais pour le W3C). Pour cela, le site doit pouvoir être consultable depuis n’importe quel type de périphériques : aussi bien sur son ordinateur de bureau que sur son PDA (après avoir synchronisé ses favoris), la page doit pouvoir s’afficher correctement. Dans le cas où l’adresse n’est pas consultable sur le périphérique mobile pour une raison précise, on doit pouvoir proposer une alternative fonctionnelle.

Le deuxième conseil, c’est d’exploiter au maximum les possibilités du périphérique afin d’améliorer l’expérience de l’utilisateur. Pour cela, il faut détecter les caractéristiques de l’appareil et lorsque plusieurs représentations sont possibles, proposer celle qui s’adapte le mieux au périphérique utilisé. Le W3C conseille donc par la même occasion de tester les pages du site sur différents périphériques ou bien à défaut, avec des émulateurs pour se rendre compte de la manière dont la page s’affiche. Néanmoins, les émulateurs ne rendent pas compte à 100% de la manière dont la page s’affiche en réalité et puis en plus, le résultat peut être différent d’une version à une autre des logiciels (OS de l’appareil, version du navigateur web, résolution de l’écran...).

Afin de faciliter la tâche de l’utilisateur et au vu de la manière de la saisie de texte sur de tels périphériques, il est demandé de raccourcir au maximum la longueur des adresses à taper pour accéder aux pages. On met en avant d’avoir recours à des méthodes pour éviter de devoir taper les adresses internet à la main (utilisation d’hyperlink, code-barre, RFID tag, ...) mais ce n’est pas toujours possible. C’est pour cela qu’on conseille d’utiliser des adresses courtes pour diminuer les erreurs possibles lors de la saisie dans un premier temps mais également, pour améliorer la satisfaction de l’utilisateur lors de sa navigation sur le site. Une des solutions proposées est l’utilisation de sous-domaine plutôt que d’avoir une page (par exemple au lieu de devoir écrire `http://www.example.org/example.html`, pouvoir écrire `http://example.org/example` et arriver sur la même page dans les deux cas).

Le W3C préconise aussi l’utilisation de barre de navigation (avec les différents liens vers les rubriques du site) en bas de la page plutôt qu’en haut comme on le trouve classiquement sur les sites web. Il faut pouvoir afficher le contenu de la page à l’utilisateur sans qu’il doive faire défiler la page (scrolling) avant de pouvoir commencer à lire le contenu. De plus, on remarque que même dans les applications développées pour les périphériques mobiles, tous les menus se trouvent en bas de l’écran plutôt qu’en haut

sur les ordinateurs. On peut aussi proposer des « accesskey » pour les liens les plus souvent utilisés. Un accesskey est un raccourci comme on en trouve dans les applications traditionnelles (Alt+O par exemple pour ouvrir un fichier dans Word) afin de ne pas devoir à chaque fois cliquer sur le lien. Néanmoins, ce procédé n'est pas des plus commodes pour un PDA étant donné qu'il n'y a pas de clavier (juste un virtuel).

Une autre recommandation importante de la part du W3C concerne encore les liens, mais cette fois, l'identification de la cible du lien. En effet, la vitesse de connexion des PDA est en général beaucoup plus lente que la connexion que l'on a à son domicile (sauf si on se trouve sur des points d'accès Wifi) et le coût est également important. En effet, en général maintenant, le coût ne se fait pas sur la durée de la connexion mais sur la quantité de données téléchargées et donc à chaque fois que l'utilisateur clique sur un lien, l'action a un coût pour ce dernier. Il est donc primordial de lui indiquer clairement le contenu qu'il va obtenir lorsqu'il va cliquer sur le lien afin qu'il soit sûr que c'est ce qu'il cherche ou qu'il a besoin. De même, lorsqu'il s'agit d'un document à télécharger, il est important de préciser le type de document dont il s'agit pour que l'utilisateur soit sûr qu'il pourra le consulter sur son périphérique et ne pas gaspiller du temps et du téléchargement pour rien.

De même, toujours concernant les liens, il est très fortement déconseillé d'avoir recours à des « image maps ». Les image maps sont des zones cliquables sur une image et dont chaque zone pointe vers un lien particulier. Prenons par exemple le cas d'une carte : on pourrait avoir une zone cliquable pour chaque région de la carte. Ainsi, les zones sont définies par des coordonnées géométriques : chaque sommet de la forme correspond à la coordonnée du point (pixel) à l'intérieur de l'image. Cet attribut est très rarement correctement interprété sur les périphériques mobiles et donc il est déconseillé d'utiliser ce genre de technique. Il est conseillé d'avoir un bouton/image par lien et qu'ils soient espacés les uns des autres.

Le W3C recommande de ne pas utiliser de « pop-up » et des scripts pour rafraîchir automatiquement la page tous les x secondes ou bien pour faire une redirection vers une autre page. Le W3C préconise d'utiliser les redirections du côté serveur avec les codes HTTP 3xx. Si on souhaite vraiment mettre en place un système de rafraîchissement de la page, on doit en informer l'utilisateur et lui donner la possibilité de le désactiver.

Enfin, pour en terminer avec la navigation et les liens, il est conseillé de limiter au maximum les liens vers des ressources extérieures du site car cela pourrait provoquer des temps de chargement beaucoup plus élevés et par la même occasion, mécontenter l'utilisateur.

## 6.4.2 Mise en page et contenu

Le contenu recherché sur un périphérique mobile et son ordinateur est différent. Sur le premier on souhaite avoir accès directement à l'information recherchée et sans avoir à lire de longs textes tandis que sur un ordinateur, on souhaite plus souvent avoir des phrases bien construites, un style littéraire plus étendu. Pour cela, on doit bien faire attention, lors de la création de son site web pour périphérique mobile à se limiter à l'information pertinente et seulement cela ! De plus, il faut bien faire attention à ne pas avoir des pages trop lourdes car la mémoire embarquée dans les périphériques mobiles est assez limitée et donc il faut que la page puisse être chargée en mémoire. Il ne faut donc pas envisager d'avoir des images, animations ou vidéo en haute qualité. Il est même conseillé d'avoir un redimensionnement automatique (et même une « dégradation » du contenu si nécessaire) pour en diminuer la taille et ainsi le temps de chargement.

Concernant les barre de défilement, il faut se limiter à une seule maximum sauf quand cela est vraiment impossible de faire autrement. Ainsi, il est préférable de privilégier le défilement vertical que l'horizontal. C'est le plus courant, même pour le site classique que l'on consulte sur un ordinateur.

On retrouve ensuite les mêmes recommandations que celles concernant l'accessibilité : utiliser des couleurs de texte et d'arrière-plan avec un contraste suffisant pour être lisible, toujours définir un texte alternatif pour les images et les animation/vidéo.

### 6.4.3 Définition des pages

Il est recommandé de définir un titre court pour chaque page mais suffisamment représentatif pour savoir le contenu que l'on pourra y trouver. Il est « interdit » d'utiliser des frames car elles ne sont pas reconnues par de nombreux navigateurs ou bien posent de nombreux problèmes lorsqu'elles sont reconnues. Elles sont également à proscrire dans le cadre d'un site web classique. Concernant la structure de la page, il est nécessaire d'utiliser les attributs du HTML pour définir les éléments de structure comme les paragraphes, les titres de section, etc.

Concernant les tables, le W3C déconseille leur utilisation mais après des tests avec une table classique et même avec des images à l'intérieur des cellules, l'affichage est le même entre notre PDA et l'ordinateur. Comme dit précédemment, il ne faut pas oublier d'ajouter un texte alternatif pour tous les éléments non-textuels et il est préférable de ne pas avoir recours à des scripts.

En ce qui concerne les images, comme nous l'avons déjà dit, il est nécessaire de bien spécifier les attributs de taille lorsqu'elle en possède une et si nécessaire de la retailler du côté server pour diminuer le temps de chargement et la mémoire nécessaire pour afficher la page en mémoire.

De plus, même si cela n'est pas valable uniquement pour le web mobile, il est nécessaire d'utiliser des marqueurs (HTML) valides et pour vérifier cela, d'utiliser les services de validation offert par le W3C par exemple.

Enfin, pour conclure les recommandations, il est préférable d'alléger au maximum les pages, en ce qui concerne les attributs afin d'accélérer le chargement. Pour cela, il est conseillé d'utiliser des CSS pour définir la présentation et de la mise en page du site. Ainsi, il est nécessaire de charger en mémoire qu'une seule fois le fichier et de l'appliquer ensuite à toutes les pages du site plutôt que de définir à chaque objet les attributs nécessaires. Concernant les cookies, il est préférable de considérer qu'ils ne sont pas activés et disponibles car c'est souvent le cas sur les périphériques mobiles et donc si ils sont vitaux pour le fonctionnement du site, alors il sera fortement probable que la navigation sur le site sera impossible. Il est également nécessaire de mettre en cache les informations dans les réponses HTTP.

### 6.4.4 Données en entrée

Dans cette section, nous allons terminer en abordant tout ce qui concerne l'entrée de données de la part de l'utilisateur. Ainsi, le W3C conseille d'obliger au minimum l'utilisateur à devoir entrer des informations libres dans des champs textes du fait de l'utilisation du clavier virtuel. Pour cela, on conseille de proposer un choix de valeur à cocher par exemple pour remplacer cela mais parfois ce n'est pas possible. Pour les mêmes raisons, il est conseillé de limiter au maximum la nécessité d'utiliser le clavier. Enfin, lorsque l'on connaît la langue, le format d'entrée du périphérique, on doit le proposer par défaut à l'utilisateur afin de lui rendre plus agréable la navigation.

## 6.5 Étude sur les requêtes écrites et orales pour l'accès à l'information dans un contexte mobile

### 6.5.1 Contexte de l'étude

Une équipe de chercheurs de l'université de Strathclyde, au Royaume-Uni, s'est intéressée aux différences entre les requêtes écrites et les requêtes orales pour accéder un contenu informationnel sur des périphériques mobiles (type GSM ou PDA). Ainsi, à travers une série d'expériences expliquées dans [26] ainsi que les résultats, l'équipe étudie l'impact du langage dans la saisie d'information. En effet, depuis assez longtemps, de nombreux travaux sont effectués dans la reconnaissance de l'écriture manuscrite mais pas dans l'usage de la parole pour la même tâche.

En moyenne, la vitesse de saisie sur les anciens téléphones portables (sans l'utilisation de dictionnaires prédictifs) est de l'ordre de 7 à 15 mots par minutes ; avec un système de saisie intuitive pour téléphones mobiles (communément appelé T9), on peut doubler cette vitesse. Avec ce constat, les chercheurs ont émis l'hypothèse que l'utilisateur pouvait rédiger une requête type de 10 mots en 20 à 40 secondes avec

la nécessité de porter son attention de manière continue durant l'opération sur le périphérique. On peut espérer doubler le nombre de mots à une vitesse comparable en utilisant un stylet pour écrire de manière manuscrite les mots (comme sur un PDA par exemple si on n'utilise pas le clavier virtuel). Néanmoins, il est fréquent que la reconnaissance ne fonctionne pas correctement dans un grand nombre de situation, surtout si nous ne sommes pas installés dans un lieu calme afin d'écrire correctement faute de quoi, les mots ne sont pas reconnus par le système.

Il faut également avouer que la recherche dans le domaine de la reconnaissance vocale à encore du travail mais dans des contextes particuliers (surtout de manière thématique), les résultats sont bons. Par contexte particulier, nous entendons un sous-langage du langage naturel : si nous prenons par exemple le domaine administratif, il est possible, assez facilement et avec d'excellents résultats, de développer un système de reconnaissance vocale et linguistique (pour plus d'informations, nous pouvons nous reporter à [24]).

L'avantage de la parole est d'être un média naturel qui ne demande pas une attention continue sur le périphérique pendant l'opération de saisie de l'information. De plus, tous les téléphones portables et PDA disposent d'un microphone qui serait utilisable pour la tâche qui nous intéresse. Les requêtes orales étant un système d'analyse en temps réel demandent plus de ressources pour y parvenir. Les chercheurs ont également remarqué que les requêtes orales étaient plus courtes et donc on y retrouve moins de contexte et de redondances. Cela a donc un impact négatif sur l'efficacité du système et le taux d'erreurs.

### 6.5.2 Présentation de l'expérience

La recherche a pour but de mener une analyse expérimentale sur un groupe d'individus potentiels dans une situation aussi proche que possible des applications du monde réel. Pour les expériences, 12 sujets ont été retenus. Pour les expériences, l'équipe a utilisé 10 topics extraits de la collection de topics du TREC<sup>5</sup>. Vous trouverez un exemple d'un topic à la figure 6.1. Les expériences se déroulent individuellement et en 2 sessions. Pour la première session, on demande au sujet de formuler une requête écrite pour les 5 premiers topics une requête orale pour les 5 derniers topics. Durant la deuxième session, qui se déroule une semaine après la première avec les mêmes sujets afin d'éviter qu'ils se souviennent des 10 topics, on inverse l'ordre des topics afin que les topics pour lesquels ils devaient formuler une requête écrite ils forment maintenant une requête orale et qu'ils forment une requête écrite pour ceux qu'ils devaient formuler une requête orale lors de la première session. À la fin de l'expérience, on interviewe pendant 10 minutes chaque participant pour préciser quelques points à propos de l'expérience et la formulation des requêtes. Pour les résultats

---

```

<id> 1
<title> Topic: Coping with overcrowded prisons
<desc> Description:
  The document will provide information on jail and prison overcrowding and how inmates
  are forced to cope with those conditions; or it will reveal plans to relieve the overcrowded
  condition.
<narr> Narrative:
  A relevant document will describe scenes of overcrowding that have become all too com-
  mon in jails and prisons around the country. The document will identify how inmates are
  forced to cope with those overcrowded conditions, and/or what the Correctional System is
  doing, or planning to do, to alleviate the crowded condition.

```

---

Figure 6.1 – Exemple d'un topic du TREC

### 6.5.3 Résultats

L'équipe de chercheurs se basent sur 3 méthodes pour collecter les résultats et les analyser :

---

<sup>5</sup>Text REtrieval Conference

- les logs du logiciel/système utilisé pour rédiger/exprimer les requêtes
- les interviews
- les questionnaires

Voici les caractéristiques des deux types de requêtes de l'ensemble des participants :

Data set	q1-q120
Number of queries	120
Unique terms in queries	328
Average query length (with stopwords)	9.54
Average query length (without stopwords)	7.48
Median query length (without stopwords)	7
Average duration	02:13

Figure 6.2 – Caractéristiques des requêtes écrites

Data set	q1-q120
Number of queries	120
Unique terms in queries	459
Average query length (with stopwords)	23.07
Average query length (without stopwords)	14.33
Median query length (without stopwords)	11
Average duration	01:58

Figure 6.3 – Caractéristiques des requêtes orales

On remarque donc que la durée moyenne d'une requête orale est plus longue qu'une écrite (avec un ratio d'environ 2.48) comme les chercheurs l'avaient pensé dans leurs hypothèses. On remarque également que les requêtes orales comportent plus de « stopwords » que les requêtes écrites. L'équipe de chercheurs avait déjà remarqué dans des expériences précédentes que si l'utilisateur n'a pas l'habitude d'interroger des sources d'informations ou d'utiliser des moteurs de recherches internet, les requêtes sont très courtes (3 mots ou moins). Avec les requêtes orales, l'utilisation de la parole encourage les participants à utiliser plus de mots. Voici un exemple de requêtes orales : "I want to find document about Grass Roots Campaign by Right Wing Christian Fundamentalist to enter the political process to further their religious agenda in the U.S. I'm especially interested in threats to civil liberties, government stability and the U.S. Constitution, and I'd like to find feature articles, editorial comments, news items and letters to the editor." tandis que l'équivalente en requête écrite est : "Right wing Christian fundamentalism, grass roots, civil liberties, US Constitution.". On peut se rendre mieux compte de cette différence sur le graphique de la figure 6.4.

Concernant la longueur des requêtes, on remarque que quelque soit le participant, la requête orale est toujours plus longue que celle écrite. Néanmoins, on remarque que chez un certain nombre de participants, la différence entre les deux est peu significative. Sur les 12 participants, on observe que 8 adoptent le langage naturel pour formuler leurs requêtes (comme une discussion orale) tandis que 4 utilisent uniquement des mots-clés pour formuler leurs requêtes. On peut trouver ces résultats sur le graphique de la figure 6.5. Ces 4 utilisateurs expliquent cela par le fait de trouver étrange et inconfortable de parler à une machine.

L'équipe a mesuré également le temps pour formuler les requêtes (un maximum de 5 minutes par topic avait été fixé pour l'expérience) et on remarque qu'en moyenne, la durée pour la requête orale est plus courte ou équivalente que pour la requête écrite. Les résultats des durées par participants ne permettent pas de dégager de tendances.

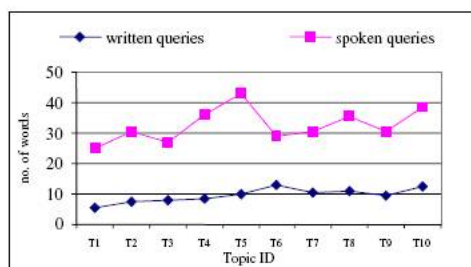


Figure 6.4 – Longueur moyenne des requêtes par topic

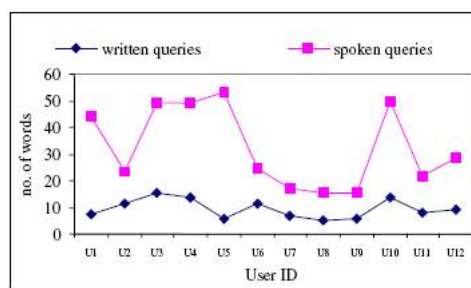


Figure 6.5 – Longueur moyenne des requêtes par utilisateur



Nous avons donc vu que les requêtes orales étaient plus longues que celles écrites. Dans les requêtes orales, les participants utilisent plus de mots comme des prépositions, des conjonctions ou des articles. Ces mots ont une certaine sémantique (même si parfois elle est faible) et permettent d'indiquer des relations grammaticales : c'est ce que la communauté de la recherche documentaire appelle « stopwords ». Dans les requêtes écrites, on sait qu'elles sont plus courtes mais contiennent des mots de contenu tel que les noms, les adjectifs et les verbes. Ainsi, sémantiquement parlant, les requêtes orales ne participent pas plus que les requêtes écrites. Après avoir supprimé les « stopwords » dans les requêtes écrites et orales, on constate néanmoins (même si le nombre de mots a diminué dans les deux cas) que les requêtes orales comportent toujours plus de mots que les requêtes écrites. Ainsi, on peut conclure que les gens ont plus de facilité à s'exprimer oralement que de manière écrite : d'une part en ce qui concerne la conversation mais également de manière sémantique. Statistiquement, on trouve une longueur moyenne de 14.33 mots pour les requêtes orales après avoir supprimé les « stopwords » tandis que pour les requêtes écrites, on trouve une moyenne de 7.48 mots.

Du point de vue de la recherche documentaire, plus on dispose de mots pour une recherche plus on améliore les résultats. Malheureusement, l'ironie est qu'en ce qui concerne la reconnaissance de la parole, le taux d'erreur moyen se situe aux environs des 10% mais ce taux d'erreur monte à 20-30% dans la reconnaissance d'une discussion avec un large vocabulaire et en ce qui un téléphone, le taux d'erreur se situe entre 30 et 40%. Ainsi, on se rend compte que l'avantage que l'on pensait d'avoir une sémantique plus riche oralement que par écrit ne permet pas d'en profiter au maximum actuellement à cause des systèmes existants. Ce qui ressort des expérimentations c'est que même si on approche un taux d'erreur de 50%, les requêtes orales ont le potentiel d'être aussi utiles que les requêtes écrites malgré la plus grande dégradation dans le sens des requêtes. On peut retrouver ces informations dans les graphiques des figure 6.6 et figure 6.7.

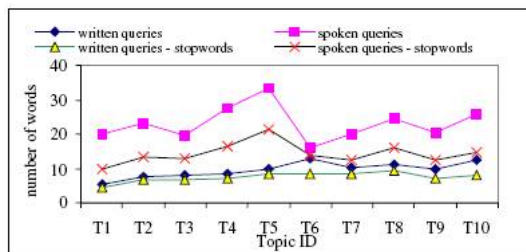


Figure 6.6 – Longueur moyenne des requêtes par topic

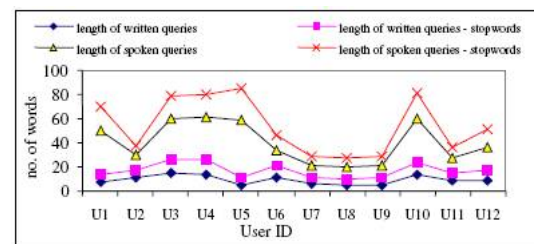


Figure 6.7 – Longueur moyenne des requêtes par utilisateur

#### 6.5.4 Conclusions

Les expériences menées par l'équipe de chercheurs a permis de montrer que les requêtes orales permettent de s'exprimer plus naturellement mais aussi encourage les personnes à exprimer plus de sémantique. Ainsi, cette méthode pour formuler et saisir des informations est parfaitement viable. Mais du fait de la taille réduite de l'échantillon de tests, on ne peut pas en généraliser les résultats. On a également remarqué l'impact du taux d'erreur dans la reconnaissance des mots oralement et c'est pour cela que dans la suite des recherches, l'équipe souhaite pouvoir en analyser les conséquences de manière détaillée et comparer les résultats aux requêtes écrites moins riches sémantiquement mais mieux reconnues par les systèmes actuels. Dans le futur, l'équipe souhaite également expérimenter cela au mandarin qui possède une structure sémantique complètement différente de l'anglais.

En ce qui nous concerne, cette analyse nous a permis de voir que les requêtes orales pourraient être une solution envisageable pour le module de recherche sur le futur portail touristique depuis les PDA.

Néanmoins, cette solution nécessite de mettre en place un système important pour la reconnaissance de la parole. L'idée est intéressante et il faudrait alors analyser en détails les coûts et les revenus que l'on pourrait tirer d'un tel système.

## 6.6 Conclusion

En nous aidant des différents documents du W3C ainsi que du livre [29], nous allons maintenant pouvoir réfléchir au layout à développer pour notre adaptation du portail aux petits écrans. Dans la section 7.3 page 85 consacrée aux recherches depuis le PDA, nous reviendrons sur une utilisation possible du son dans la formulation des requêtes par l'utilisateur.

## 7.1 Idées générales de la présentation

Maintenant que nous avons vu les principes généraux et les recommandations des spécialistes du domaine des IHM, il nous faut réfléchir sur la manière dont nous pourrions les mettre en pratique pour nos fiches touristiques.

Dans un premier temps, nous allons simplement réfléchir sur la présentation générale du contenu des fiches à l'écran. Par la suite, nous réfléchirons sur la manière de visualiser chaque élément à l'écran. Il sera peut-être nécessaire à ce moment-là de revoir légèrement notre mise en page mais nous n'y sommes pas encore.

### 7.1.1 Propositions de layout

Il a été recommandé par le W3C d'éviter autant que possible d'avoir deux barres de défilement simultanément, il faut choisir entre une verticale ou une horizontale. Étant donné la configuration d'un PDA où l'écran est plus haut que large (même si sur les derniers modèles plus perfectionnés il est possible d'inverser le sens de l'écran), il est préférable d'opter pour une barre de défilement verticale. Il faudra donc avoir une page de maximum 240 pixels de large afin de ne pas avoir de scroll horizontal. De plus, il a été recommandé plusieurs fois, pour diverses raisons que l'on retrouve dans la section 6.4 page 62, de ne pas avoir des pages trop lourdes. Afin d'y parvenir, nous devons donc diviser les informations en plusieurs rubriques qu'il nous faut définir. Après, pour la représentation, plusieurs solutions s'offrent à nous : nous pourrions penser à un système d'onglets où chaque onglet représente une rubrique ou bien alors un système de fragmentation de l'information (il existe des algorithmes pour exécuter cette tâche automatiquement) et ensuite les afficher sous forme d'une arborescence avec des + et des - (comme sous l'explorateur de fichiers de Windows par exemple) permettant d'afficher ou de cacher l'information du niveau donné. La deuxième solution permet donc de suivre la recommandation du W3C préconisant d'éviter d'avoir de trop longs scrolls à effectuer pour lire la page mais néanmoins, cela n'empêche pas d'avoir des pages trop lourdes car le contenu de chaque rubrique caché, même s'il n'est affiché, est chargé en mémoire : cela nécessite donc un temps de chargement plus long et une plus grande quantité de mémoire nécessaire. Pour remédier à ce problème, il faudrait être capable de ne charger, au démarrage de la page, que les niveaux de l'arborescence (où chaque niveau correspond à une rubrique) et de ne charger ensuite le contenu de la rubrique que lorsque l'utilisateur clique sur le + pour en afficher le contenu. Néanmoins, un tel système ne doit pas être simple à développer et surtout, pas forcément compatible avec les versions mobiles des applications (comme le navigateur). De plus, si nous optons pour une telle solution, la page

de la fiche touristique serait assez vide et pas vraiment attractive pour l'utilisateur. Il ne peut en effet pas lire les informations principales sans devoir cliquer sur une rubrique ce qui pourrait frustrer à terme un grand nombre d'utilisateurs. Nous privilégierons donc la première solution avec un système d'onglets.



Figure 7.1 – Layout basé sur une barre de navigation

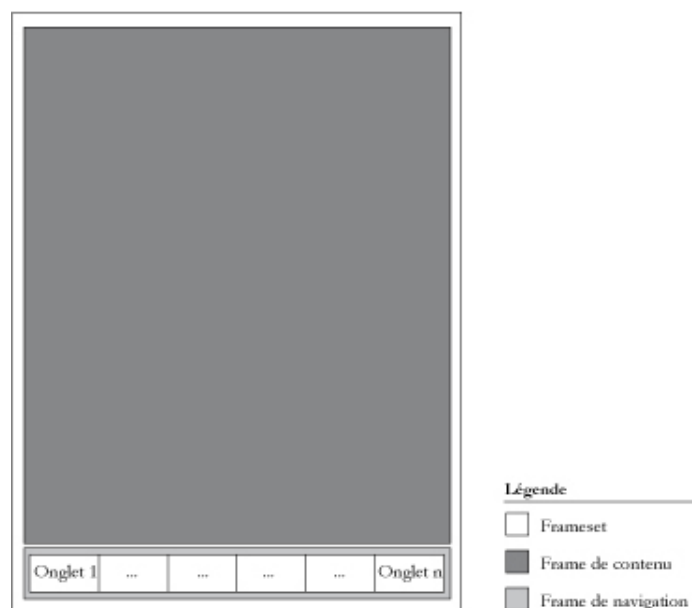


Figure 7.2 – Layout basé sur un frameset

Pour être en accord avec les recommandations du W3C de [53], nous afficherons la barre d'onglets en bas de l'écran comme nous pouvons le voir sur le schéma figure 7.1. Le problème que l'on peut avoir c'est si la page est plus haute que l'écran, l'onglet ne serait pas visible immédiatement. Il faudrait alors soit en haut de la page mettre un lien pointant vers le bas de la page et la barre d'onglets ou bien essayer de trouver une solution pour afficher la barre d'onglets en bas de l'écran tout en permettant de faire défiler le texte dans la partie haute de la fenêtre.

On pourrait donc utiliser un système de cadres : un cadre pour le contenu informationnel en haut de la page et le barre d'onglets dans le deuxième cadre, en-dessous du premier comme on peut le voir sur le schéma figure 7.2. Mais on a vu précédemment que les frames posent beaucoup trop de problèmes d'accessibilité et qu'elles ne sont pas gérées par les navigateurs web des périphériques mobiles. Il faut donc abandonner cette idée. Il existe sinon, en passant par un CSS, des instructions qui permettent de positionner à un endroit fixe de notre choix sur l'écran, des éléments. Malheureusement, comme nous le savons, Microsoft et son navigateur web Internet Explorer ne respectent pas toujours les standards web du W3C ; avec la version 7, les instructions sont correctement interprétées mais pas avec les versions précédentes. Il risque donc d'y avoir certains problèmes avec Pocket IE installé sur les PDA. Avec Firefox ou Opera, cela fonctionne correctement. Malgré cela, on peut retenir cette proposition de layout qui semble la plus adaptée pour respecter les recommandations du W3C et rappeler les applications des périphériques mobiles. Nous allons donc poursuivre notre analyse en se basant sur ce layout (figure 7.1).

Maintenant que nous avons défini la mise en page de nos fiches touristiques, nous allons réfléchir à la manière d'organiser nos rubriques.

### 7.1.2 Réflexions sur les rubriques des fiches

Il faut maintenant réfléchir, en se basant sur la DTD du fichier XML ainsi que le schéma ERA, les différentes thématiques qui ressortent afin d'en faire ensuite des groupes. Les grandes thématiques qui en ressortent, sans s'occuper d'un ordre de classement, sont les suivantes : coordonnées, présentation générale, horaires/périodes d'ouverture, détails pratiques, équipements/services offerts (uniquement pour les objets d'hébergement et de restaurant), répartition des chambres/logements (uniquement pour les objets d'hébergement). Détaillons à présent les attributs repris dans chacune de ces thématiques. Pour cela, nous avons repris le schéma ERA en colorant d'une couleur différente chaque attribut en fonction de la thématique à laquelle il appartient. Voici la légende des couleurs :

- rouge pour la présentation générale
- bleu pour les coordonnées
- orange pour les horaires/périodes d'ouverture
- vert pour les détails pratiques
- violet pour les équipements/services offerts
- gris pour la répartition des chambres/logements

On retrouve le schéma avec la coloration des thèmes à la figure 7.3 page suivante.

Maintenant, il faut ordonner les thématiques pour l'affichage graphique. L'ordre choisit nous paraît être logique et naturel. Le voici :

1. présentation générale
2. coordonnées
3. horaires/périodes d'ouvertures
4. détails pratiques
5. équipements/services offerts
6. répartition des chambres/logements.

Il est possible que nous soyons encore obligés de le modifier par la suite, mais en principe c'est dans cet ordre que nous souhaitons afficher les informations. Il faudra ensuite définir l'ordre d'affichage des éléments à l'intérieur de chaque groupe mais nous pensons qu'il est préférable d'analyser la représentation graphique de chaque élément avant. En effet, suivant la représentation choisie, l'ordre pourra changer. Nous allons donc maintenant réfléchir et analyser la représentation graphique de chaque élément et ensuite, nous reviendrons sur la question de l'ordre de chaque élément à l'intérieur de chaque groupe thématique.

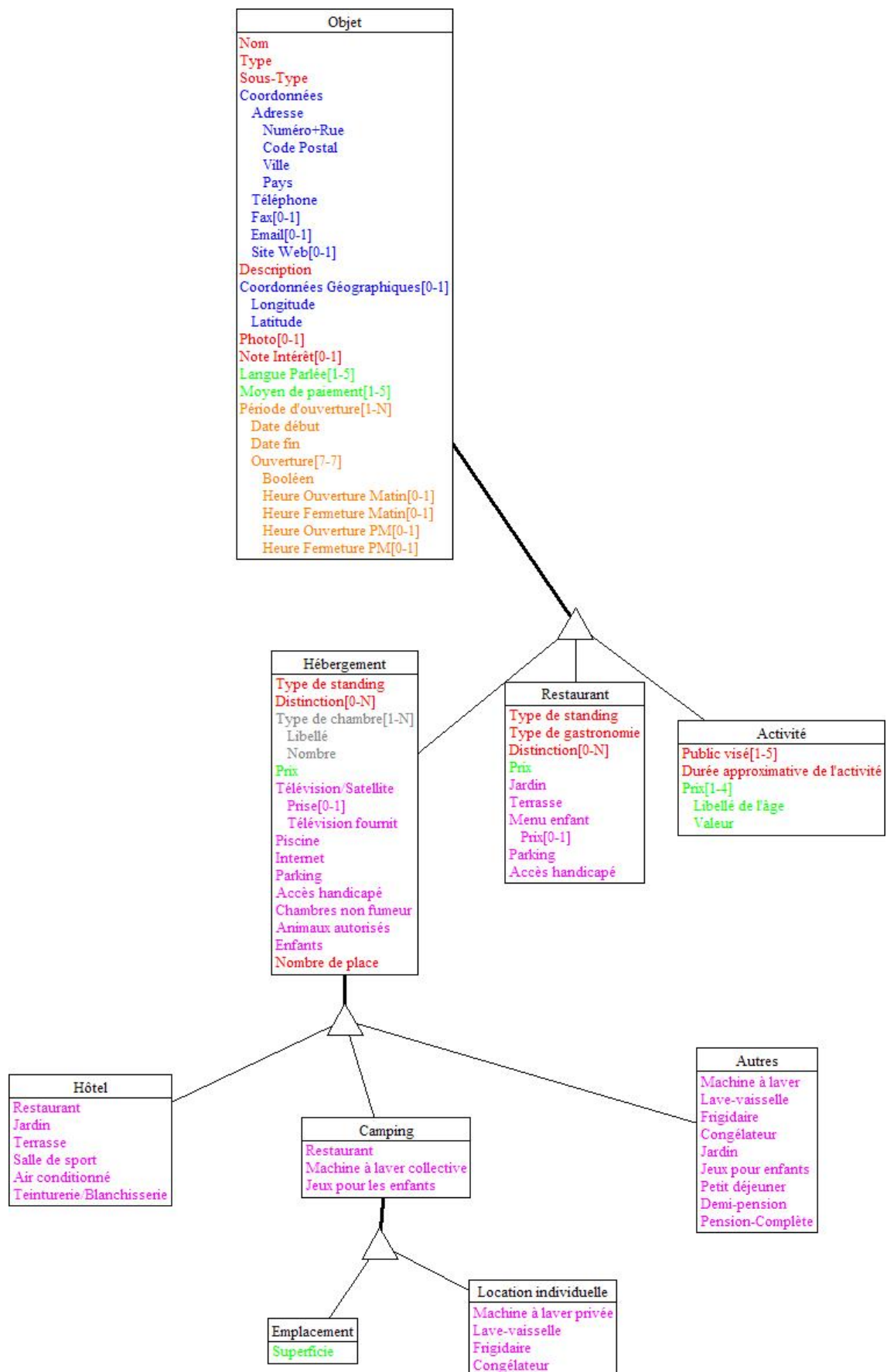


Figure 7.3 – Thématiques des informations des fiches touristiques

## 7.2 Représentation graphique conceptuelle des interfaces

Afin de pouvoir s'abstraire au maximum de tous les problèmes liés à un choix de matériels précis ainsi que d'anticiper l'évolution technologique des périphériques mobiles assez rapide, il est important de définir les interfaces et la navigation d'un niveau plus élevé sans s'occuper de l'implémentation et des outils graphiques utilisés par la suite afin de pouvoir facilement réutiliser tout le travail d'analyse préalable ainsi que les réflexions sur la représentation graphique pour un autre support ou bien après une évolution technologique très importante des PDA actuels sans devoir tout reprendre à zéro. C'est toute cette réflexion sur la représentation graphique sous forme d'une « représentation graphique virtuelle » que l'on va trouver dans cette partie.

Étant donné la nature de l'information à représenter et du domaine d'application qui est le tourisme, on est tenté de vouloir presque tout représenter sous la forme d'icônes associées à une légende. Malheureusement, ceci n'est pas très original et n'améliore pas ce qui existe déjà. Il est donc nécessaire de réfléchir d'avantage à des solutions innovantes tout en restant fonctionnelles et facilement compréhensibles par les utilisateurs.

L'utilisation d'icônes pour la visualisation sur PDA est possible grâce à l'amélioration des écrans et de la qualité d'affichage. En effet, au début des PDA avec des écrans en noir et blanc et des résolutions très faibles, il aurait été impossible d'utiliser des icônes pour représenter de l'information. Maintenant, les derniers modèles ont des écrans VGA avec une résolution de 640x480 pixels. Il s'agit donc d'une qualité proche des premiers écrans d'ordinateurs dont certains utilisateurs utilisent encore pour leur ordinateur de bureau un peu ancien. Ce système d'icônes est donc une solution adéquate pour la représentation d'une partie des informations. Il ne faudra pas également oublier de prévoir sur la fiche touristique, un bouton pour ajouter l'objet au panier de l'utilisateur, si cette fonctionnalité est développée par la suite (cf le paragraphe 7.4 consacré au panier page 90).

Avant de commencer notre réflexion de visualisation possible, voici un exemple de fiche du site des gîtes de France<sup>1</sup> disponible à la figure 7.4.

Gîte n°136113 à GRAND-BORNAND



Classement : 

Capacité : 9 personnes

Nombre de chambres : 3


Surface : 87 m<sup>2</sup>

Altitude : 1000 m

Orientation : Sud-Ouest

Animaux non acceptés

Thème :  
Gîte bébé

 album photo







[réserver](#) [voir les dispo.](#)

**LA BRODERIE N°2 :** Entre chaîne des Aravis et village traditionnel, chalet neuf comprenant 3 gîtes et le logement des propr. Gîte de 87 m<sup>2</sup> aménagé au 1er niveau (jardin de plain pied). Cuis. équipée (m-onde) ouverte sur grand séjour (dic-dac 2 pers.), poêle à bois (foyer apparent), Ch1 (1 lit 160), Ch2 (3 lits 1 pers.), Ch3 (1 lit 160, lit bébé), bains/vc, sèche linge, salle d'eau, wc. Chauff. central. Terrasse, jardin, balcon, garage. Gîte situé à l'entrée de la Vallée du Bouchet, au pied des pistes de fond. Domaine skiable 800 m. Toutes activités et commerces à 1.2km. Bois fourni, pos. loc<sup>e</sup> draps/linge de maison et toilette. Orient<sup>e</sup> Sud/Ouest.

**Equipements :**  
Garage, Rez de chaussée, Lave-linge, Télévision, Lave-vaisselle, Cheminée, Terrain, Lit enfant, Sèche-linge, Micro-ondes, Chauffage fioul

**Distances :**

											
0.8	SP	30.0	1.2	1.2	SP	6.0	40.0	1.2	1.2	32.0	1.2

SP : sur place

Figure 7.4 – Exemple d'une fiche du site des gîtes de France

<sup>1</sup><http://www.gites-de-france-haute-savoie.com>

Nous avons également consulté les idées développées dans les articles suivants [21] et [23] afin d'avoir quelques idées concernant la représentation d'informations sur des petits écrans, et dans les deux cas présents, des informations statistiques. Cela nous a permis de voir qu'il était possible d'afficher clairement une quantité importante d'informations en utilisant un jeu de couleurs adaptés et des méthodes de visualisations adéquates. C'est ce que nous allons essayer de proposer pour le domaine touristique qui nous intéresse dans ce travail.

### 7.2.1 Représentation graphique du nom

Le nom de l'objet (hébergement, restaurant ou activité) n'est pas l'élément principal de la fiche même si évidemment on doit le voir et le connaître. De plus, il n'y a pas énormément de manière possible pour le représenter : soit simplement de manière textuelle (avec un corp de texte plus grand et peut-être un effet de style pour le mettre en évidence) ou bien sous forme d'une image avec par exemple le logo de l'objet si il existe. A part ces deux possibilités, dans l'immédiat, il ne nous vient rien à l'esprit comme autres solutions envisageables.

### 7.2.2 Représentation graphique des coordonnées géographiques

Les coordonnées géographiques qui commencent à être utilisées dans des sites internet comme celui de Michelin permettent d'offrir un certains nombres de services tel que la localisation du lieu sur une carte afin d'avoir une idée générale de sa localisation, permettre d'effectuer l'itinéraire pour s'y rendre, obtenir la météo, etc... Actuellement seule la première possibilité est utilisée alors il est vraiment intéressant de creuser dans ce domaine afin d'offrir de nouvelles possibilités surtout si l'utilisateur est sur la route. On pourra donc utiliser différentes API<sup>2</sup> disponibles auprès de sites comme celui de Michelin, de Google (en particulier Google Maps) ou bien celui de l'IGN qui est en train de se développer et s'étoffer. Le problème sera de récupérer les coordonnées géographiques de chaque objet. Ainsi notre objectif est d'offrir une localisation sur carte que l'on affichera sur la page des coordonnées ainsi que des services de propositions d'itinéraires de promenades et pourquoi pas des informations météorologiques si cela est possible (par l'intermédiaire de Web Services par exemple).

### 7.2.3 Représentation graphique des coordonnées

Les coordonnées de contact de l'objet composées de l'adresse complète (numéro, rue, code postal, ville et pays) ainsi que du numéro de téléphone, de fax, url du site de l'objet si il existe ainsi que l'email. À notre avis, comme pour le nom, il n'existe pas vraiment de solutions originales pour les représenter. On associera à cela, mais nous y reviendrons par la suite, différents éléments cartographiques liés aux coordonnées géographiques de l'objet. Mais concernant la représentation graphique, nous pensons que l'idéal est la forme textuelle avec la possibilité d'utiliser des icônes devant le numéro de téléphone, fax, email afin de bien les mettre en valeur. Ce qui pourra être aussi intéressant de faire, c'est en fonction de la taille de l'information à représenter, utiliser un procédé pour ne pas afficher toute l'information en même temps mais simplement par exemple l'adresse, ou bien les coordonnées téléphoniques ou bien les informations internet ainsi, cela permettrait peut-être, dans le cas où la quantité d'information est important, d'éviter d'avoir une barre de défilement vertical. Nous allons donc combiner les coordonnées avec les coordonnées géographiques. Nous retrouvons à la figure 7.5 page suivante le layout que nous proposons. Dans le premier cadre, nous retrouvons les coordonnées/informations de contact de l'objet touristique : le nom/titre, l'adresse, le téléphone/fax avec un symbole pour symboliser l'information, enfin l'email et l'url du site web si il existe. À droite du cadre, on retrouve un emplacement pour indiquer la météo actuelle (ou les prévisions météorologiques) sur le lieu de l'objet touristique. Ce service serait rendu possible grâce à un web service/API d'un site comme Météo-France, le service météo de Yahoo, etc. En dessous, on trouve une carte issue de Google Maps par exemple ou bien plus tard issue du géoportail de l'IGN quand le système sera complètement mis en place. Cette carte permettra de situer l'objet

---

<sup>2</sup>Application Programming Interface



afin que l'utilisateur puisse le trouver plus facilement. Si l'utilisateur clique sur la carte, cela ouvrirait l'application portable de Google Maps si elle est installée sur le périphérique portable de l'utilisateur afin de pouvoir utiliser les bénéfices d'un GPS si l'utilisateur en possède un, ou bien calculer un itinéraire pour se rendre vers l'objet touristique, etc. Enfin, on trouve en bas de l'interface, une rubrique **Vous aimerez aussi** qui proposent à l'utilisateur 3 autres objets touristiques qu'il pourrait apprécier en rapport avec l'objet consulté actuellement. Ce système serait basé sur l'ontologie du portail touristique développé par Antidot/Mondeca. On retrouve également un lien **Et bien d'autres encore** qui lancerait une recherche en rapport avec l'objet actuel et proposer plus de résultats que les 3 affichés sur la fiche.

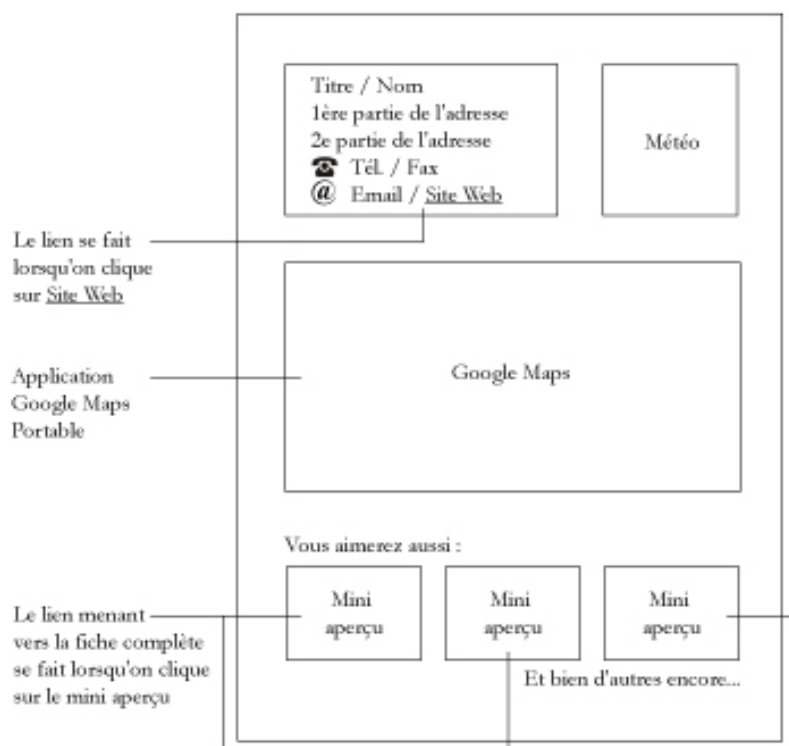


Figure 7.5 – Layout pour les coordonnées de l'objet

#### 7.2.4 Représentation graphique de la description

La description est une information purement textuelle qui doit être insérée de cette manière dans la fiche. La seule nouveauté à laquelle nous pensons, c'est d'utiliser le son également pour cette information. Ainsi, l'utilisateur pourra écouter la description en même temps qu'il peut la lire ou bien regarder d'autres informations en même temps. En effet, la description est un élément qui se couple parfaitement avec le son comme on commence à le voir de plus en plus avec les livres audio et depuis longtemps avec les audio-guides dans les musées ou sites touristiques.

On trouve à la figure 7.6 page suivante, un exemple de présentation possible pour l'onglet concernant la **présentation générale** de l'objet.

Le champ 1 de la figure 7.6 page suivante représente le logo symbolisant le type d'objet touristique (un restaurant, un hébergement ou une activité). Le champ 2 de la figure 7.6 page suivante représente la note d'intérêt de l'objet qui est détaillée à la section 7.2.18 page 84. On trouve évidemment le texte de présentation de l'objet touristique et enfin un bouton afin d'ajouter l'objet en cours de consultation dans son panier afin de pouvoir le retrouver rapidement lors d'une prochaine consultation du portail touristique.

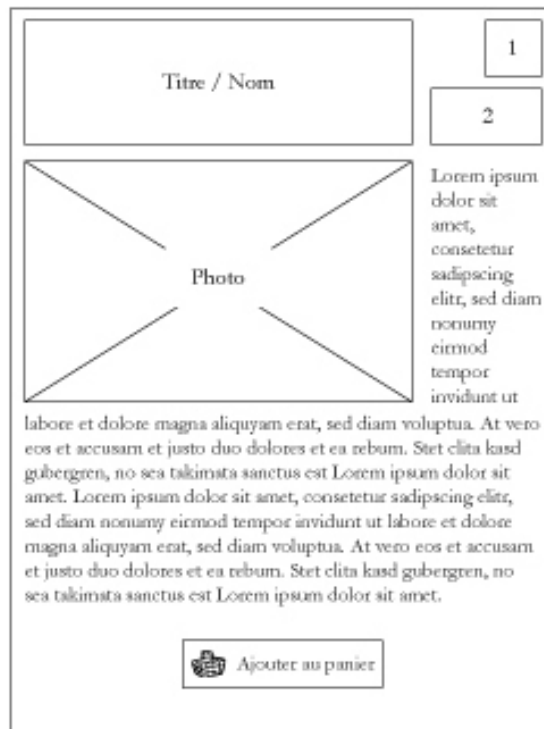


Figure 7.6 – Layout pour la présentation générale de l’objet

### 7.2.5 Représentation graphique des photos/vidéos

La représentation graphique de photos et/ou vidéos dans la fiche touristique se fera classiquement comme pour un site sur ordinateur de bureau avec une image de taille réduite sur la page de la fiche et un lien vers une taille plus élevée. On pourra si on le veut ajouter une bordure graphique autour de l’image ou de la vidéo. Ce qui pourrait être intéressant de mettre en place, c’est un système de diaporama pour visualiser automatiquement toutes les photos les unes à la suite des autres. On pourrait agrémenter cela de brefs commentaires audio afin de présenter ce qui se trouve sur la photo lorsque cela est nécessaire (inutile par exemple pour les photos de l’intérieurs des chambres d’un hôtel ou d’un gîte par exemple). Nous utiliserions le format jpeg pour les photos, concernant les vidéos, étant donné la difficulté d’en afficher facilement sur une page web (même sur un navigateur web classique, nous avons besoin de plugins), il serait plus facile d’utiliser le format Flash (flv) puisqu’il existe une version mobile du plugin flash (version 7 actuellement). Pour l’audio, nous utiliserions des formats classiques comme le wav ou le mp3 qui sont gérés par n’importe quel PDA.

Voici un exemple de ce que l’on entend par diaporama et que l’on pourrait implémenter pour PDA :



Figure 7.7 – Exemple de diaporama dynamique

### 7.2.6 Représentation graphique des horaires et périodes d'ouverture

Actuellement, sur les sites existants (pas spécialement ceux du tourisme), aucun effort n'a été fait concernant les horaires et périodes d'ouverture. Tout est textuel et nous devons lire le paragraphe en entier pour savoir si l'objet est ouvert le jour où l'on souhaite y aller et quelles sont les heures d'ouverture. Il faut donc faire un effort pour améliorer les choses dans ce domaine. L'idéal est d'avoir un script qui permet de dessiner une horloge (numérique ou analogique) en lui passant simplement l'heure et les minutes en argument. Une solution plus simple est d'avoir un fichier graphique (gif ou jpeg par exemple) pour chaque horaire différent et d'afficher celui qui convient mais c'est une solution plus lourde à mettre en place. Enfin, une solution combinant l'avantage de chaque proposition est d'avoir une image au format jpeg de l'horloge et de la placer en background du « DIV » de la page web dans lequel on insère ensuite l'heure sous forme de texte. Ainsi, nous n'avons plus besoin d'implémenter un script pour cette tâche. Concernant les périodes d'ouverture, on pourrait utiliser un système de calendrier par exemple et utiliser une couleur différente pour l'ouverture et la fermeture. Suivant la solution retenue pour les périodes d'ouverture (un calendrier complet de l'année), il pourra être judicieux d'avoir recours à un système de regroupement afin de ne pas avoir besoin d'afficher l'année complète à l'écran.

Avant de proposer une représentation complète des horaires et périodes d'ouverture, voici le moyen utilisé par les gîtes de France pour indiquer la disponibilité d'un hébergement. On peut donc se baser sur ce genre de modèle pour indiquer les périodes d'ouverture (utilisé un code de couleur défini par une légende pour indiquer pour chaque jour l'ouverture/fermeture et l'horaire). On pourra aussi se baser sur cela pour ensuite, indiqué le nombre de places disponibles pour les hébergements, à la section 7.2.15 page 83. La capture du site se trouve à la figure 7.8 page suivante.

Voici maintenant une idée de présentation de l'onglet **horaires/périodes d'ouvertures** que l'on trouve à la figure 7.9 page suivante.

L'idée est donc d'avoir un calendrier pour chaque mois et d'attribuer pour chaque plage horaire d'ouverture une couleur et donc en légende, d'indiquer les heures d'ouvertures pour le matin et l'après-midi.

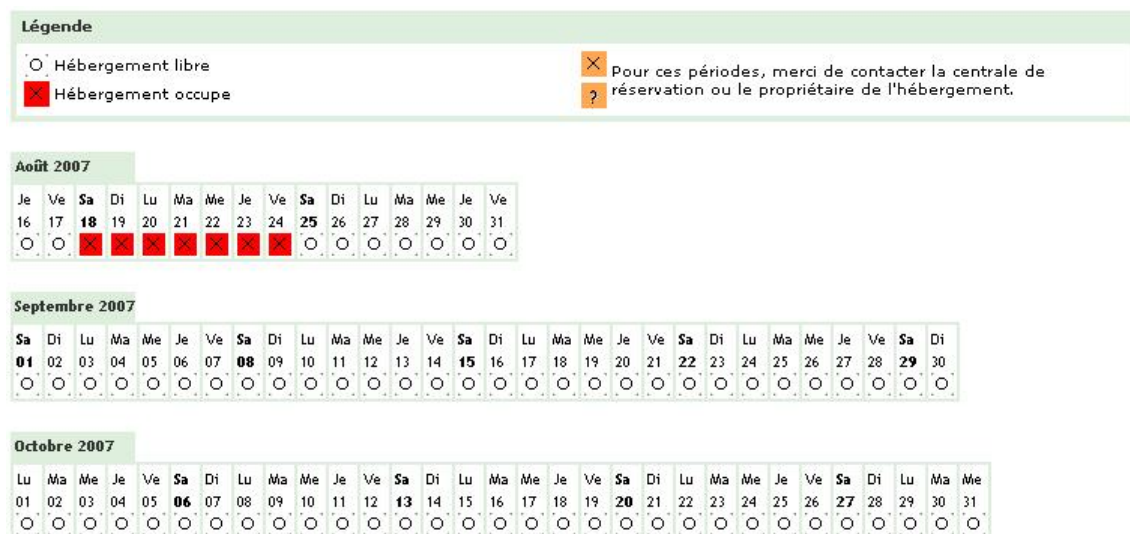


Figure 7.8 – Système de disponibilité d'un hébergement des gîtes de France

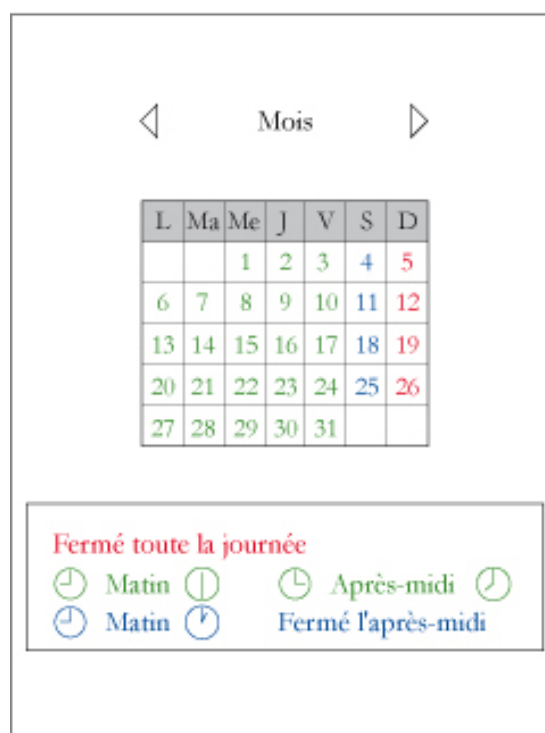


Figure 7.9 – Layout pour les horaires et les périodes d'ouvertures

### 7.2.7 Représentation graphique des prix, des tarifs des activités

Pour les hébergements ou les restaurants, les prix sont indiqués sous la forme d'un intervalle avec le prix minimal pour la valeur gauche et le prix maximal pour la valeur droite. On a valeur droite > valeur gauche. Il existe néanmoins deux cas particuliers :

- pour indiquer un prix « jusqu'à x€ », on aura un intervalle  $[0 ; x]$
- pour indiquer un prix « à partir de x€ », on aura un intervalle  $[x ; -1]$  et donc cette fois valeur droite < valeur gauche mais il s'agit d'un cas bien particulier et puis de toute manière, il est impossible d'avoir un prix négatif donc -1 a été choisi comme valeur extrême.

On pourra utiliser une règle pour indiquer la fourchette des prix avec des zones de couleurs allant du plus clair au plus foncé (blanc pour les petites valeurs et rouge voir noir pour les plus grandes : blanc → jaune → rouge → noir) afin d'avoir rapidement une idée de la fourchette des prix et en indiquant sur la règle, la valeur du prix minimal et du prix maximal numériquement afin d'avoir les valeurs précises.

On trouve une illustration de cela à la figure 7.10. Le premier exemple illustre le cas d'un intervalle de prix entre 50 et 75€ (l'intervalle est symbolisé par un rectangle et les prix sont indiqués en plus de manière écrite). Le deuxième exemple illustre le cas d'un prix à partir de 50€ (symbolisé par un intervalle semi-fermé à gauche) et enfin le dernier exemple illustre le cas d'un prix jusqu'à 50€ (symbolisé par un intervalle semi-fermé à droite) avec à chaque fois la valeur écrite afin de pouvoir connaître le prix d'un seul regard.

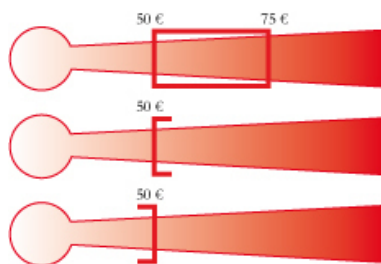


Figure 7.10 – Layout pour les prix

Concernant les activités, les prix sont définis différemment car il peut, en effet, y avoir un tarif différent en fonction de l'âge. Pour cela, chaque tarif est un couple de valeur (libellé, prix) où le libellé indique la catégorie de l'âge et le prix, la valeur de l'entrée pour l'activité. On pourra associer chaque libellé à un symbole afin d'alléger l'affichage textuel. Pour le prix, on pourra réutiliser le même principe que pour les hébergements et les restaurants.

### 7.2.8 Représentation graphique des moyens de paiement

La représentation graphique des moyens de paiement (espèce, chèque, Visa, Mastercard, ...) est un élément important pour l'utilisateur car cela lui permet de vérifier qu'il pourra facilement payer dans l'établissement qu'il choisit. Actuellement, cette information est représentée sous forme iconographique d'un spécimen de taille réduite de la carte ou du moyen de payer. Après réflexion, à l'heure actuelle, nous ne voyons pas comment on pourrait représenter autrement cette information... Ce qui pourrait être intéressant et utile pour l'utilisateur d'ajouter, qui n'est pour le moment, jamais présent : l'intervalle de prix où l'on peut utiliser chaque moyen de paiement. Ainsi on pourra savoir si la Visa n'est acceptée qu'à partir de 20€ par exemple ou bien l'espèce uniquement jusqu'à 50€, etc... afin de pouvoir s'organiser à l'avance. Malheureusement, cette information n'est pas connue ou que très rarement et il serait très difficile de l'implémenter. Si nous décidions de l'implémenter, nous devrions définir un intervalle de prix, en réutilisant les notations et les principes exposés pour les prix et tarifs de l'objet, dans la section 7.2.7. Il faudrait alors modifier la structure de fichier XML par l'intermédiaire de la DTD ainsi que les fichiers XML d'exemples. Ainsi, pour le moment, nous partons du principe que nous n'implémenterons pas.

terons pas cette fonctionnalité : nous afficherons uniquement les différents moyens de paiement supportés.



Figure 7.11 – Visa



Figure 7.12 – Mastercard



Figure 7.13 – Chèque



Figure 7.14 – Espèce

## 7.2.9 Représentation graphique des services/équipements offerts

Pour la représentation graphique des services et équipements offerts, nous pensons qu'on pourra utiliser un système d'icônes (une pour chaque élément) mais étant donné le nombre important d'éléments possibles, on risque de se retrouver avec un nombre important d'icônes à retenir et comprendre pour l'utilisateur.

On pourrait également utiliser un tableau avec une colonne par élément et une croix par exemple pour indiquer si le service est offert ou non.

Reprenons une fiche du site des gîtes de France et intéressons-nous à la rubrique **Distances**. Cette rubrique reprend différentes activités principales que cherchent les vacanciers (piscine, tennis, équitation, gare...) et indique la distance séparant l'hébergement de l'activité. Pour cela, ils ont utilisé un pictogramme pour chaque activité qu'ils ont disposés dans un tableau avec en-dessous de chacun, un chiffre pour la distance. Nous pensons donc reprendre cette idée mais en remplaçant le chiffre par une croix pour indiquer que le service/équipe est offert par l'hébergement et sinon un trait pour indiquer qu'il ne l'est pas. On trouve l'exemple d'une fiche de gîtes de France à la figure 7.15. À la figure 7.16, nous trouvons un exemple de ce que nous souhaitons proposer avec quelques icônes d'exemples. On pourra rajouter celles que nous souhaitons avoir dans notre fiche.

**Distances :**

0,8	SP	30,0	1,2	1,2	SP	6,0	40,0	1,2	1,2	32,0	1,2

SP : sur place

Figure 7.15 – Distances entre différentes activités et l'hébergement











									
X	X	-	X	-	X	X	-	X	X

Figure 7.16 – Services/équipements offerts

### 7.2.10 Représentation graphique du type de standing

Comme pour les précédents éléments, il est tout à fait possible d'utiliser des icônes différentes pour indiquer le standing comme on le retrouve actuellement dans les guides actuels (que ce soit sur papier ou Internet) mais il est intéressant de chercher une autre solution. Nous suggérons, un peu comme pour les intervalles de prix, un système de barre de défilement, avec un système de couleur associée (pourquoi pas aussi plusieurs icônes pour indiquer des paliers) afin de savoir en un instant, sans avoir besoin de connaître la signification des icônes de quel type d'établissement il s'agit.

### 7.2.11 Représentation graphique des distinctions, labels

En ce qui concerne les distinctions et les labels, nous pensons que malheureusement il n'existe pas d'alternative pour leur représentation graphique. En effet, chaque label touristique a son logo ainsi que pour les distinctions (étoiles du guide Michelin, la toque du Gaultmillau, épis des gîtes de France...) et donc il est vraiment plus logique de réutiliser ces logos sous forme de pictogrammes.



Figure 7.17 – Gaultmillau



Figure 7.18 – Gîtes de France



Figure 7.19 – Michelin

### 7.2.12 Représentation graphique du type de gastronomie

Concernant la représentation graphique du type de gastronomie, nous pensons qu'une bonne solution serait d'afficher le drapeau du pays (par exemple le drapeau français pour de la gastronomie française, le drapeau italien pour un restaurant italien ou une pizzeria, etc...) et par exemple pour un établissement de restauration rapide (type snack, ...) on pourra utiliser le drapeau américain qui est à l'origine de ce genre de gastronomie et d'ajouter un hamburger sur l'icône en plus.. On pourrait également, en plus, indiquer un commentaire textuel pour préciser les informations si cela est nécessaire. La figure 7.20 représente un exemple pour un restaurant servant des plats de gastronomie française traditionnelle.

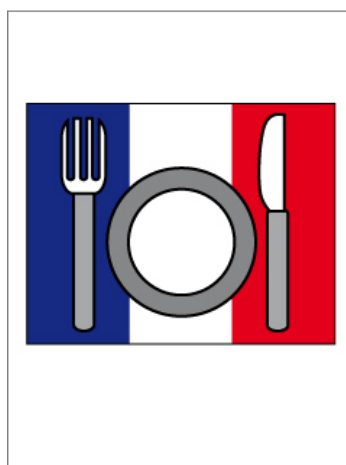


Figure 7.20 – Type de gastronomie

### 7.2.13 Représentation graphique du public visé

Concernant le public visé, l'idée qui vient à l'esprit est d'utiliser une icône différente pour chaque catégorie d'âge (une pour les enfants, une pour les adolescents, une pour les adultes) ou bien une signalétique différente pour chaque catégorie d'âge comme on trouve pour les livres, les jeux vidéos, etc. Sinon, en conservant cette idée, on pourrait améliorer le système en affichant pour chaque catégorie d'âge une note ou bien un moyen pour indiquer l'adéquation de la catégorie pour l'activité. Ainsi par exemple, si on a une activité qui est plutôt réservée aux enfants, on pourrait avoir 4 étoiles pour les enfants, 2 pour les adolescents et par exemple 1 pour les adultes. L'utilisateur pourra donc savoir à l'avance, lorsqu'il fait ses recherches et son planning, organiser le séjour en fonction des membres de la famille. Dans le cas où il y a une activité **interdite** à une catégorie d'âge, on pourra indiquer à la place des étoiles, un panneau sens interdit de la circulation routière par exemple pour indiquer qu'en plus de ne pas plaire à cette catégorie d'âge, on ne peut pas leur permettre de la pratiquer (par exemple dans le cas d'activité sportive trop dangereuse pour les petits enfants).

Comme le système d'étoiles est souvent associé à l'idée de standing pour les restaurants ou les hôtels, nous avons décidé d'associer à chaque catégorie un pictogramme (un enfant en vert, un adolescent en bleu et un adulte en rouge). Ensuite, pour symboliser l'adéquation, on pourra afficher de 0 à 4 pictogrammes pour la catégorie d'âge où 0 indique que l'activité n'est pas adaptée et 4 signifie que l'activité s'adresse en priorité pour cette catégorie d'âge. Un exemple est disponible à la figure 7.21. Sur l'exemple, nous n'avons pas représenté d'interdiction pour une catégorie d'âge mais nous pourrions par exemple utiliser l'idée proposée dans le paragraphe précédent.



Figure 7.21 – Public visé

### 7.2.14 Représentation graphique de la durée de l'activité

Pour représenter la durée de l'activité, nous pensons à une petite horloge analogique et au milieu, à la place des aiguilles, indiquer la durée approximative de l'activité sous la forme HH :MM comme nous pouvons le voir sur la figure 7.22 page suivante qui indique que la durée moyenne de l'activité est de 1h45.



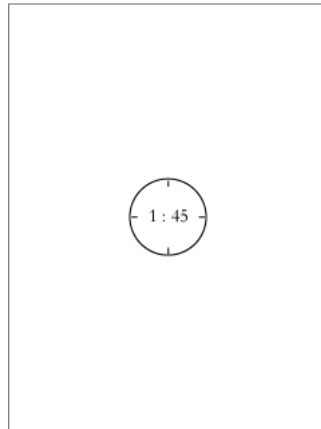


Figure 7.22 – Durée d’une activité

### 7.2.15 Représentation graphique du nombre de places totales disponibles

Nous pensons que pour la représentation du nombre de places disponibles, il peut y avoir de nombreuses choses de faites. Tout d’abord un affichage simple et efficace (sûrement textuellement) le nombre total de places disponibles dans l’hébergement afin d’éviter par exemple pour une famille de 6 personnes d’obtenir des résultats à la recherche avec des hébergements pour 4 personnes.

Par la suite, il pourrait être possible aussi d’indiquer le nombre de places restantes suivant les périodes par l’intermédiaire d’histogrammes par exemple avec comme intervalle la semaine et une vue mensuelle afin d’éviter d’avoir un graphique trop grand qui serait illisible sur les PDA. Il faudrait donc évidemment que le système soit automatique (avec donc un web service entre le site touristique et l’hébergement) pour connaître en temps réels le nombre de places disponibles. Cette deuxième partie ne pourra pas être implémentée durant ce stage et dans le cadre du mémoire, mais un travail conceptuel pourra être néanmoins fait. Par exemple un nombre écrit en texte pour indiquer le nombre total et ensuite on peut préciser par un histogramme le nombre de places disponibles par catégories. On trouve une représentation du résultat à la figure 7.23.

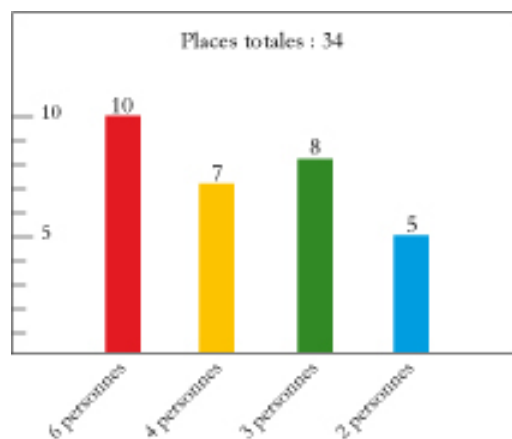


Figure 7.23 – Nombres de places totales disponibles

### 7.2.16 Représentation graphique de la répartition des chambres par type

La répartition des chambres par types se présentera sous la forme d'un camembert afin d'avoir une vue globale compréhensible d'un seul coup d'œil, sans devoir lire de longs tableaux. Dans chaque part du camembert, on trouve le nombre de chambres pour un type de chambre. L'information expliquant le type de chambre dont il s'agit (2 pers. avec une douche, 2 pers. avec une baignoire, etc.) figure dans la légende. Le résultat se trouve à la figure 7.24.

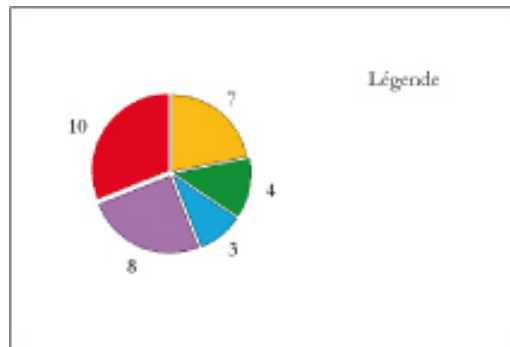


Figure 7.24 – Répartition des chambres par type

### 7.2.17 Représentation graphique de la superficie des emplacements

Concernant la superficie des emplacements de camping, nous pensions faire un peu sur le même principe que la répartition des chambres par types pour les hôtels. En effet, les emplacements de camping n'ont pas forcément la même taille et pas forcément les mêmes installations (branchement eau, électricité, etc...) donc on pourrait adapter la DTD et donc également les fichiers XML pour gérer ces informations et pouvoir ensuite, dessiner un camembert pour montrer graphiquement la répartition des emplacements selon leurs types (emplacement pour tente, emplacement pour caravane, groupe de superficie, équipements disponibles, etc.). On retrouve donc le même principe que celui utilisé pour la répartition des chambres par type et donc la visualisation est la même que la figure 7.24.

### 7.2.18 Représentation graphique de la note d'intérêt

Comme il a été indiqué dans le paragraphe concernant la sémantique des éléments, il s'agira d'une note par exemple compris entre 0 et 5 et il faudra donc trouver une icône pouvant indiquer clairement de quoi il s'agit et ne se confondant pas avec d'autres (comme par exemple les distinctions). On pense qu'il est bien d'utiliser un cœur plein rouge par exemple. Afin que l'utilisateur puisse voir d'un coup d'œil la note maximum, nous pensons qu'il faut toujours afficher les 5 cœurs et choisir la couleur rouge pour celle qui composent la note et en gris celle qui ne la compose pas. Voici un exemple pour mieux comprendre : une note de 3/5 serait donc représenter avec 3 cœurs rouges et 2 cœurs gris (ainsi l'utilisateur voit directement que la note maximale est 5 et se donne une idée de la qualité et de l'intérêt de l'objet) ; pour une note de 0/5, on aurait 5 cœurs gris et pour une note de 5/5, 5 cœurs rouges. Si nous n'affichons pas en permanence le nombre maximum d'icônes, l'utilisateur ne peut pas savoir sur combien l'objet est noté : 3/5 est une note moyenne tandis que si c'était 3/10, cela devient vraiment très mauvais. La figure 7.25 page suivante représente ainsi une note de 2/5.

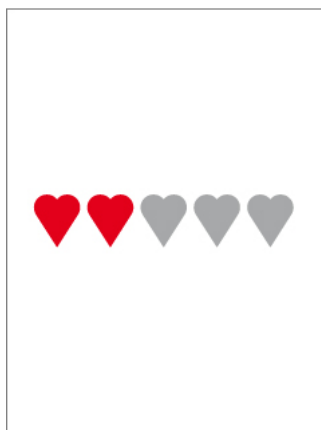


Figure 7.25 – Note d'intérêt

### 7.3 Représentation graphique des requêtes de recherche

Après avoir réfléchi à la représentation graphique de la fiche des objets, il est maintenant temps de s'intéresser à la saisie des requêtes de recherche de manière graphique. En effet, sur un PDA, avec un stylet et un clavier virtuel, il n'est pas aisé de saisir des requêtes « à la main » et donc il est nécessaire de trouver une autre solution, basée sur des éléments graphiques afin d'éviter au maximum de devoir écrire des mots pour effectuer sa recherche. On pourrait également avoir recours à la parole pour formuler les requêtes comme nous en avons discuté à la section 6.5 page 64 mais c'est un système qu'il nous est impossible d'expérimenter pour ce mémoire. Nous avons donc décidé de nous limiter aux requêtes de recherche graphiques. Il est donc nécessaire de développer des scénaris d'interactivité afin de pouvoir développer une interface en adéquation avec ces derniers.

On pense que le premier écran de la saisie de la requête doit proposer la/les carte(s) géographique(s) de plus haut niveau concernée par l'offre touristique du portail. Par « plus haut niveau » nous entendons un pays ou bien une région administrative traitée par le portail touristique : il pourrait s'agir par exemple de la carte d'un pays entier comme la France ou alors une ou plusieurs régions comme l'Île-de-France ou bien un ou plusieurs départements comme Paris. L'utilisateur pourra alors sélectionner une ou plusieurs zones géographiques qui l'intéressent afin d'affiner sa recherche. En fonction du niveau actuel, il est possible que l'utilisateur puisse restreindre la zone en sélectionnant de nouvelles zones plus petites en fonction de ses choix précédents. Par exemple si au niveau précédent il avait choisi la région de la Bourgogne, le nouvel écran lui permettrait de choisir soit l'entièreté de la région ou bien un ou plusieurs départements parmi les 4 qui composent la région : la Côte-d'Or, la Nièvre, la Saône-et-Loire et l'Yonne. On proposera ainsi un niveau supplémentaire tant qu'il existe un découpage plus restreint des zones choisies par l'utilisateur ou tant qu'il n'a pas sélectionné l'entièreté de la zone pour sa recherche. Ainsi sur chaque écran, il y aura une ou plusieurs cartes géographiques de la zone concernée et l'utilisateur pourra cocher les zones qui l'intéressent.

L'étape suivante consistera à choisir le type d'objet recherché (hébergement, activité, restaurant). Pour cela, nous pensons qu'il faudra avoir 3 images représentant chacune des possibilités et l'utilisateur n'aura qu'à cliquer sur celle qui l'intéresse comme le montre la figure 7.26 page suivante.

Dans l'étape suivante, il faudra demander à l'utilisateur de saisir ses préférences ou priorités dans ce qu'il recherche (confort, sport, repos, high-tech, culture...) en fonction du type d'objet qu'il aura précédemment sélectionné (il faudra donc définir cette liste avant de commencer l'implémentation) que l'on



Figure 7.26 – Choix du type d'objet touristique recherché

pourra ensuite être utilisé pour lui proposer des résultats en accord avec ses attentes (afin d'éviter par exemple de lui proposer de nombreuses activités sportives si l'utilisateur déteste le sport par exemple). Graphiquement, nous pensons offrir la possibilité à l'utilisateur de facilement classer l'ordre de ses priorités. Ainsi, il aura la liste dont on vient de parler et ensuite des flèches pour descendre ou monter l'élément. Ce système permettra donc à l'utilisateur, de rapidement classer ses préférences et au portail de lui proposer des restaurants, hébergements ou activités suivant ce qu'il préfère. On aurait également pu développer un système de représentation par un camembert et l'utilisateur pouvait définir un pourcentage pour chaque thème mais il est toujours assez difficile de quantifier quelque chose et en plus d'avoir un total de 100 pour l'ensemble. Nous gardons donc la première solution représentée à la figure 7.27 page ci-contre.

Ensuite, on pourra demander à l'utilisateur la composition de la famille (nombre d'adultes, d'enfants, d'adolescents) afin de pouvoir au mieux lui proposer des activités ou des hébergements qui pourraient convenir à tous. Si au départ de la recherche l'utilisateur émet le choix de chercher des restaurants, il ne sera pas nécessaire de lui afficher cet écran. Nous pensons utiliser un champ de saisie numérique où l'on peut incrémenter/décroémenter la valeur. Il faudra un champ par catégorie de personnes comme nous pouvons le voir à la figure 7.28 page suivante.

Ensuite la recherche s'exécute et il reste à afficher les résultats. Si l'utilisateur fait une recherche pour un restaurant ou un hébergement, l'écran suivant lui permettra de préciser sa recherche : comme par exemple le type de restaurant (snack rapide, restaurant gastronomique, étoilé, ...) ou d'hébergement (camping, hôtel, palace, gîte, ...) ainsi que des critères comme les équipements proposés et un intervalle de prix. En ce qui concerne la recherche d'une activité, l'écran lui permettra de définir le type d'activité qu'il recherche : sportive, culturelle ou manuelle. C'est la liste de types que nous avons défini lors de l'analyse et la structuration des fiches touristiques à la section 5 page 39. Il est possible par la suite évidemment d' étoffer cette liste.


Enfin, dans le dernier écran, il faudra autoriser l'utilisateur à entrer des mots-clés pour préciser la re-

Classez suivant vos priorités :


Confort
Sport
Repos
Nouvelles technologies
Culture

Figure 7.27 – Classement de préférences de l'utilisateur par priorité


Composition de la famille :



0



0



2

Figure 7.28 – Encodage de la composition de la famille

87

cherche. Il faudrait également pouvoir incorporer à cette étape ou bien dans un écran supplémentaire, le système développé par Mondeca<sup>3</sup> basé sur des ontologies afin d'ajouter de la sémantique dans la recherche ainsi que pour aider l'utilisateur dans ses choix. Sur la version classique du portail de la Nièvre, l'ontologie est affichée en permanence sous forme d'arborescence sur le côté de l'écran. Du fait de la petite taille des écrans de PDA, il est impossible de faire la même chose. C'est pour cela que l'on garde le système mais qu'on l'utilise qu'à la fin de la requête pour l'affiner au maximum. Après une étude des logs de navigation des utilisateurs mobiles (en se basant sur le navigateur utilisé) et si l'utilisation de l'ontologie est importante, nous pourrions rajouter un mode de recherche à part (à choisir dès le départ entre recherche graphique et recherche avec l'ontologie). Il nous est néanmoins impossible de mener une telle étude puisque la fin du développement du projet est prévu pour juillet 2009 et de voir si il est utile ou non de proposer cette fonctionnalité pour la version portable du site.

L'idéal serait d'offrir la possibilité à l'utilisateur de revenir à l'étape précédente de la recherche afin de modifier ses critères sans devoir tout recommencer depuis le début (de la saisie). De plus, afin de retenir les différentes informations entrées par l'utilisateur (la composition de la famille, ses préférences, etc.), on pourrait lui proposer avant de commencer sa recherche de se connecter à son compte (panier) afin de charger les informations déjà connues et de lui demander si il souhaite les recharger ou bien entrer des informations différentes. On pourra aussi, comme nous le détaillerons à la section 7.4 page 90, proposer à l'utilisateur de sauvegarder les recherches qu'il a effectué afin de pouvoir les réexécuter facilement et rapidement.

## 7.4 Affichage des résultats d'une recherche

Maintenant que nous avons défini la manière de représenter graphiquement les éléments d'une fiche touristique, de la manière d'effectuer une recherche (et les éléments graphiques que l'on souhaite utiliser), il faut définir la manière dont on souhaite afficher les résultats à l'utilisateur. Dans sa thèse [14], Nicolas Bonnel développe une partie sur la restitution des résultats de recherche ainsi que son organisation. Son travail s'intéresse aux ordinateurs classiques mais certains concepts ou idées peuvent néanmoins être intéressants et utiles même si par la suite, il a essentiellement travaillé sur des présentations interactives en 3 dimensions, chose actuellement impossible pour différentes raisons sur un PDA. Bénédicte Le Grand, même si elle ne s'est pas intéressée en particulier aux résultats de recherches, a également travaillé dans sa thèse [38] sur la représentation de systèmes complexes sémantiquement structurés, chose à laquelle nous sommes arrivés grâce à notre structure de fiches touristiques en XML que l'on pourrait par la suite associer à des ontologies comme l'a fait Mondeca sur le site de la Nièvre et comme on souhaite y parvenir dans le projet Eiffel [10].

Il faudra donc s'aider de ces différents travaux tout en gardant à l'esprit que cela doit être parfaitement utilisable sur un PDA (qui possède un petit écran et une basse résolution). On pourra également s'intéresser à la manière dont les résultats sont affichés sur le site de la Nièvre qui est classique mais efficace. Il faudra sûrement revoir le contenu des informations qui résumeront chaque fiche mais la présentation est claire et parfaitement adaptable aux PDA ; ce qui est une bonne chose vu qu'on essaye malgré tout de rester proche du site de la Nièvre, pour notre site mobile étant donné que nous ne savons pas vers quoi le projet Eiffel aboutira du point de vue de l'interface graphique. Actuellement, sur le site de la Nièvre, on affiche les résultats les uns en-dessous des autres avec le nom de l'objet, une photo, une icône pour indiquer le type de l'objet et le début de la description. Il faudra juste voir sur le PDA, si il est possible de mettre la photo car il faudra sûrement diminuer la taille de l'image par rapport à la version actuelle et elle risque de devenir trop petite pour être vraiment utile (si elle est trop petite et que l'on ne voit rien, autant ne pas la mettre). Sur le site de la Nièvre, toutes les fenêtres intermédiaires s'ouvrent dans la

---

<sup>3</sup>L'utilisation de l'ontologie est une des principales nouveautés offert par l'outil de recherche touristique et le web sémantique

même à l'exception de la fiche touristique qui s'ouvre dans une nouvelle fenêtre plus petite (du même type qu'un pop-up). Le problème est que le W3C recommande formellement de ne pas utiliser de pop-up sur des périphériques mobiles. Les nouveaux navigateurs web (Opera Mobile ou Mozilla Minimo) pour PDA supportent les onglets (comme sur les versions pour ordinateurs des navigateurs web) et donc ce problème devient moins gênant. Ces remarques considérées, il serait judicieux d'afficher la fiche touristique dans une nouvelle fenêtre (qui se caractériserait par un nouvel onglet du navigateur web) afin d'éviter de devoir utiliser la fonction « précédent » du navigateur qui parfois, perd le contenu de la page précédente. Le problème est qu'il est plus difficile de passer d'une fenêtre à une autre sur un PDA que sur un ordinateur classique et qu'il n'existe pas encore les onglets sur tous les navigateurs (principalement Pocket IE installé par défaut sur tous les Pocket PC). Il existe des logiciels spécifiques payant qui permettent parfois d'ajouter de nouvelles fonctionnalités par rapport à Pocket IE mais ils ne sont sûrement pas utilisés par la majorité des utilisateurs et Opera Mobile est payant ; il reste néanmoins Mozilla Minimo qui est gratuit.

Concernant le contenu à afficher, nous pensons que les éléments suivants sont nécessaires :

- le nom
- la note d'intérêt
- le type de standing
- le type de gastronomie
- le début de la description (200-300 caractères maximum)<sup>4</sup>
- photo si la réduction de la taille la rend toujours compréhensible et utilisable.

On trouve un exemple de représentation à la figure 7.29. Suivant le type d'objet, tous les champs ne seront pas visibles. En effet, le type de standing ne concerne que les restaurants et les hébergements tandis que le type de gastronomie uniquement les restaurants.

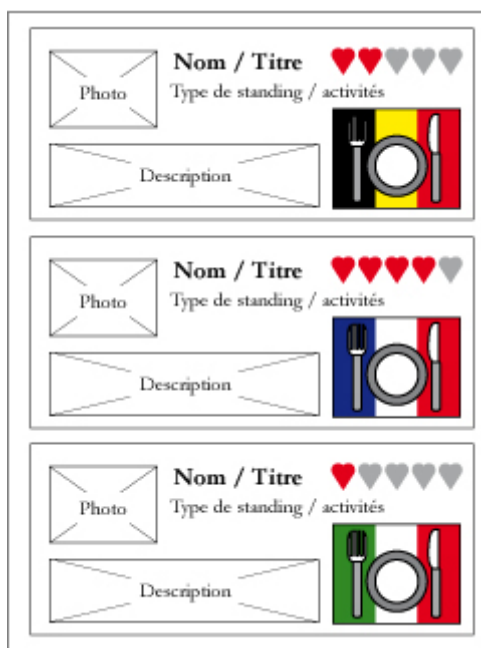


Figure 7.29 – Écran d'affichage des résultats d'une recherche

De plus, il est fréquent que tous les critères émis lors de la requête ne puissent pas être satisfaits de la même manière. En effet, il peut arriver qu'un ou plusieurs critères ne puissent pas être vérifiés

<sup>4</sup>On pourrait également ajouté un champ spécifique pour une description courte dans les fiches XML car les premiers caractères de la description ne sont pas forcément significatifs.

complètement. Afin de l’afficher clairement à l’utilisateur, nous pourrions avoir recours à un diagramme en étoile où chaque branche de l’étoile correspond à chaque critère de la requête. Ce genre de graphique est de plus en plus utilisé auprès du grand public pour résumer graphiquement certains résultats (test d’un appareil par exemple dans certaines revues comme Test-achats) ou bien dans des jeux vidéos « à la mode » comme ceux d’entraînements du cerveau (où l’on affiche les catégories où la personne est la plus douée : logique, géométrie, ...). Un exemple d’un tel diagramme se trouve à la figure 7.30. Sur les branches, nous avons choisi de voir figurer les données suivantes :

- Préférences personnelles (confort, sportif, repos, high-tech, culturelle...)
- Composition de la famille
- Prix
- Lieu géographique
- Type d’objet (camping, gîte, restaurant gastronomique, palace, etc.)

qui sont caractéristiques des éléments principaux de la recherche possible du portail. Ce graphique serait affiché à côté de chaque résultat afin que l’utilisateur puisse juger si il souhaite le consulter ou non.

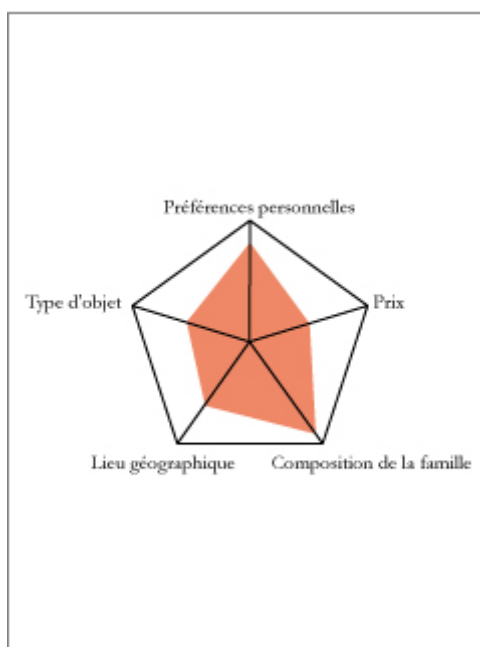


Figure 7.30 – Graphique en étoile

Ensuite, un clic sur le nom ou bien la photo (si on l’affiche) permettra d’ouvrir la fiche complète de l’objet.

#### 7.4.1 Représentation graphique du panier

L’analyse et la réflexion concernant la représentation graphique du panier est vraiment très difficile car il s’agit d’un élément qui n’existe pas actuellement et dont nous ne disposons pas actuellement d’informations sur les fonctionnalités qui seront développés. Ce que nous pouvons donc proposer, c’est d’offrir la possibilité à l’utilisateur de consulter son panier créé depuis son ordinateur et enregistré sur le portail touristique mais il ne pourra pas ajouter des éléments ou bien créer un nouveau panier ou compte utilisateur. Il s’agira seulement d’une consultation des éléments mémorisés afin de pouvoir retrouver les coordonnées du musée, de l’hôtel ou du restaurant qu’il avait repéré auparavant.



## 7.5 Implémentation possible des fiches touristiques

Nous avons choisi d'utiliser des fichiers XML pour stocker les informations des fiches touristiques afin d'avoir une certaine liberté dans nos choix d'implémentation et le standard que constitue XML.

Le même système est actuellement utilisé pour le site de la Nièvre par Mondeca. Pour l'affichage des fiches touristiques (qui sont des fichiers XML), ils ont recours à un script XSL et des feuilles de styles (CSS) et donc on peut facilement changer l'affichage et la mise en page de la page HTML sans devoir modifier chaque fichier sur le serveur. Si on désire modifier le layout des fiches et de l'affichage des résultats, nous n'avons que le script XSL et le fichier CSS à modifier et rien de plus. On aurait le même genre de possibilités si nous avions opté pour le stockage des fiches dans une base de données et que l'on utiliserait un langage comme PHP (ou bien ASP ou des servlets Java) mais cette fois, le code d'affichage se serait trouvé dans la base de données, en tout cas pour le PHP.

Le problème avec un tel système, c'est qu'il nécessite un serveur Web et les bons packages correctement configurés pour pouvoir tester le fonctionnement. Malheureusement, les installations sont chez Mondeca et Antidot et nous n'y avons pas accès pour faire des tests de ce genre.

On trouvera donc en annexe un exemple de fiche touristique (fichier XML) de chaque type (hébergement, restaurant et activité). Le script XSL utilisé pour l'affichage des fiches du site de la Nièvre ainsi que les résultats des recherches sur un navigateur web classique est présent sur la version électronique de ce mémoire par faute de place en annexe. Nous avons également ajouté le CSS ainsi que le code HTML d'une fiche pour les périphériques mobiles.

Nous n'avons pas pu tester nos scripts XSL donc nous nous contenterons des figures de ce à quoi ressembleraient les fiches touristiques sur petit écran.

Une des premières choses à faire avant la mise en place de l'adaptation du portail touristique aux périphériques mobiles serait le déploiement et le test du script de conversion des fichiers XML en pages Web.



Dans le cadre du projet RNTL Eiffel, nous avons essayé de réfléchir à la manière d'adapter le système de portail sémantique pour le domaine du tourisme, en cours de développement par différents partenaires, aux petits écrans qui commencent à prendre de plus en plus d'importance dans notre quotidien. Le projet étant dirigé par des partenaires privés, Mondeca et Antidot, nous n'avons pas pu avoir accès au front et au back office gérant la plateforme du portail touristique que l'on retrouve à la figure 1.1 page 4 et à la figure 1.2 page 4. C'est en particulier l'accès au moteur de recherche et d'indexation d'Antidot et de Mondeca reposant sur des ontologies que nous avons expliqué à la section 2.1.6 page 12 qui a posé problème. C'est la raison pour laquelle nous n'avons pas pu réutiliser pleinement ce concept novateur dans notre adaptation aux petits écrans du portail thématique.

De plus, du fait de la longueur importante du projet Eiffel, qui s'étale sur 3 ans, nous ne pouvions pas connaître le layout retenu pour les interfaces du site classique afin de nous baser dessus pour mener à bien notre travail. Il nous a donc été nécessaire de partir de zéro et d'analyser l'existant dans le domaine touristique, tout support confondu, pour mener notre réflexion. Le résultat que nous proposons sera donc peut-être complètement différent de celui auquel le projet Eiffel aboutira pour la version classique du portail.

Après avoir analysé le contenu des fiches actuels que l'on retrouve sur le portail de la Nièvre, nous avons été obligés de définir une structure pour nos fiches touristiques afin d'avoir un contenu suffisamment riche pour ensuite travailler sur la visualisation. Pour cela, nous nous sommes appuyés sur ce qui existe actuellement (les guides touristiques au format papier, les sites web comme celui de Michelin, etc.) et nous nous sommes également interrogés sur ce qu'il manquait et l'information que nous souhaiterions trouver en tant que touristes. Ainsi, grâce à cela, nous avons pu définir, dans la section 5 page 39, une structure sémantique pour nos fiches d'objets touristiques.

Ensuite, nous avons effectué des recherches sur l'accessibilité et l'ergonomie des interfaces, en particulier pour le web et le web mobile. Cela nous a ainsi permis de proposer un layout pour les fiches touristiques et une visualisation adaptée pour les informations contenues à l'intérieur ainsi que pour la formulation des requêtes de recherches.

Néanmoins, il reste des possibilités pour approfondir le travail et offrir de nouvelles fonctionnalités pour l'utilisateur. Comme nous l'avons abordé à la section 6.5 page 64, il serait très intéressant de s'orienter vers la formulation orale des requêtes et donc de s'intéresser au domaine de la linguistique et de la reconnaissance vocale. De plus, à l'heure actuelle, la notion de panier souhaitée au sein du projet Eiffel

est vague et peu élaborée. Pour cela, nous avons dû restreindre notre réflexion sur ce point et n'avons pu proposer à l'utilisateur de la version mobile que des fonctionnalités limitées. Ces dernières sont la mémorisation d'une liste d'objets qu'il trouve intéressant et la consultation des objets mémorisés. On pourrait penser à un système plus complet comme le propose l'office du tourisme de la Nouvelle-Zélande et même de l'améliorer comme nous en avons parlé dans la section 4.2.1 page 34. On pourrait ainsi proposer un outil de planification de son séjour, de pouvoir effectuer automatiquement les réservations nécessaires pour l'hébergement, le transport, le restaurant, etc. tout en offrant une certaine validation du planning : par exemple prévenir l'utilisateur que l'objet qu'il a sélectionné pour tel jour est fermé ou bien que l'activité ne convient pas à tous les membres de la famille, ...

- [1] Xml. W3C. .
- [2] *Workshop on Personalisation on the Semantic Web PerSWeb05*, 2005.
- [3] Cartes topiques. Wikipedia France, Août 2006. [http://fr.wikipedia.org/wiki/Ontologie\\_\(informatique\)](http://fr.wikipedia.org/wiki/Ontologie_(informatique)).
- [4] tolog : Language tutorial. Ontopia, Novembre 2006. <http://www.ontopia.net/omnigator/docs/query/tutorial.html>.
- [5] Logique de description. Wikipedia France, Février 2007. [http://fr.wikipedia.org/wiki/Logique\\_de\\_description](http://fr.wikipedia.org/wiki/Logique_de_description).
- [6] Ontologies (informatique). Wikipedia France, Mars 2007. [http://fr.wikipedia.org/wiki/Topic\\_Maps](http://fr.wikipedia.org/wiki/Topic_Maps).
- [7] Web sémantique. Wikipedia France, Janvier 2007. [http://fr.wikipedia.org/wiki/Web\\_sémantique](http://fr.wikipedia.org/wiki/Web_sémantique).
- [8] Gregory D. ABOWD, Christopher G. ATKESON, Jason HONG, Sue LONG, Rob KOOPER et Mike PINKERTON : Cyberguide : a mobile context-aware tour guide. *Wirel. Netw.*, 3(5):421–433, 1997.
- [9] Alfio ANDRONICO, Antonella CARBONARO, Luigi COLAZZO, Andrea MOLINARI, Marco RONCHETTI et Anna TRIFONOVA : *Designing Models and Services for Learning Management Systems in Mobile Settings*, volume 2954 de *Lecture Notes in Computer Science*, pages 90–106. Springer Berlin / Heidelberg, 2004.
- [10] ANTIDOT et MONDECA : Projet RNTL 2005 Eiffel : Web Sémantique et e-Tourisme. Rapport technique, 2005.
- [11] Marie Aude AUFAURE, Bénédicte LE GRAND, Michel SOTO et Nacera BENNACER : *Metadata- and Ontology-Based Semantic Web Mining*, chapitre IX, pages 259–295. Idea Group Publishin, 2006.
- [12] Melanie AURHAMMER, Peter HANAPPE et Luc STEELS : *Augmenting Navigation for Collaborative Tagging with Emergent Semantics*, volume 4273 de *Lecture Notes in Computer Science*, pages 58–71. Springer Berlin / Heidelberg, 2006.
- [13] Somnath BANERJEE, Arobinda GUPTA et Anupam BASU : *Online Transcoding of Web Pages for Mobile Devices*, volume 2795 de *Lecture Notes in Computer Science*, pages 271–285. Springer Berlin / Heidelberg, 2003.
- [14] Nicolas BONNEL : *Génération dynamique de présentations interactives en multimédia 3D, de données, pour des applications en ligne*. Thèse de doctorat, Université de Rennes 1, 2006.
- [15] Christian BORNTÄGER, Keith CHEVERST, Nigel DAVIES, Alan DIX, Adrian FRIDAY et Jochen SEITZ : *Experiments with Multi-modal Interfaces in a Context-Aware City Guide*, volume 2795 de *Lecture Notes in Computer Science*, pages 116–130. Springer Berlin / Heidelberg, 2004.

- [16] Jonathan P. BOWEN et Silvia FILIPPINI-FANTONI : *Personalization and the Web from a Museum Perspective*, pages 63–78. Pittsburgh, Mars-Avril 2004. <http://www.archimuse.com/mw2004/papers/bowen/bowen.html>.
- [17] Peter BRUSILOVSKY et Riccardo RIZZO : *Accessing Web Educational Resources from Mobile Wireless Devices : The Knowledge Sea Approach*, volume 2954 de *Lecture Notes in Computer Science*, pages 54–66. Springer Berlin / Heidelberg, 2004.
- [18] Tarak CHAARI et Frédérique LAFOREST : Génération et adaptation automatique des interfaces utilisateurs pour environnements multi-terminaux projet sefagi : Simple environnement for adaptable graphical interfaces. *Ingénierie des Systèmes d'Information*, X(X):1 à X, 2002.
- [19] Tarak CHAARI, Frédérique LAFOREST et Augusto CELENTANO : Adaptation in context-aware pervasive information systems : The secas project. *Journal of Pervasive Computing and Communications*, 2(2):1 à 14, Juin 2006.
- [20] Thippaya CHINTAKOVID : Cultural differences and end-user computing. In *VLHCC '05 : Proceedings of the 2005 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'05)*, pages 325–326, Washington, DC, USA, 2005. IEEE Computer Society.
- [21] L CHITTARO, Noirhomme M. et Randolet F. : Visualization of annual time series on pdas. In Julie A. Jacko V. KATHLENE LEONARD, éditeur : *Human Computer International 2005, U.S. CD*, volume 4 de *Theories Models and Processes in HCI*. HCI International 2005, Las Vegas, Lawrence Erlbaum Associates, Inc, Juillet 2005.
- [22] Todd CORAM et Jim LEE : Experiences – a pattern language for user interface design.
- [23] Grégory CUSTINNE, Monique NOIRHOMME-FRAITURE et Luca CHITTARO : Visualisation d'informations boursières sur téléphones mobiles. In *IHM 2004 : Proceedings of the 16th conference on Association Francophone d'Interaction Homme-Machine*, pages 215–218, New York, NY, USA, 2004. ACM Press.
- [24] Guy DEVILLE : INFO 2327 Introduction aux méthodes et concepts essentiels de l'ingénierie linguistique. Syllabus, 2007.
- [25] R. Stanley DICKS : Mis-usability : on the uses and misuses of usability testing. In *SIGDOC '02 : Proceedings of the 20th annual international conference on Computer documentation*, pages 26–30. ACM Press, 2002.
- [26] Heather DU et Fabio CRESTANI : *Spoken versus Written Queries for Mobile Information Access*, volume 2954 de *Lecture Notes in Computer Science*, pages 67–78. Springer Berlin / Heidelberg, 2004.
- [27] Mark DUNLOP, Alison MORRISON, Stephen MCCALLUM, Piotr PTASKINSKI, Chris RISBEY et Fraser STEWART : *Focussed Palmtop Information Access Combining Starfield Displays with Profile-Based Recommendations*, volume 2954 de *Lecture Notes in Computer Science*, pages 79–89. Springer Berlin / Heidelberg, 2004.
- [28] Ari FELDMAN : Web site interface design theory : A designer's primer. Rapport technique, Flying Yogi.
- [29] Wilbert O. GALITZ : *The Essential Guide to User Interface Design*. Wiley Computer Publishing, 2e édition, 2002.
- [30] Thierry GILLAIN : Intégration de données issues du web. Mémoire de Maîtrise, Facultés Universitaires Notre-Dame de la Paix - Namur, 2006.
- [31] Cathal GURRIN, Alan F. SMEATON, Hyowon LEE, Kieran McDONALD, Noel MURPHY, Noel O'CONNOR et Sean MARLOW : *Mobile Access to the Fischlar-News Archive*, volume 2954 de *Lecture Notes in Computer Science*, pages 124–142. Springer Berlin / Heidelberg, 2004.
- [32] James HOBART : Principals of good gui design. Rapport technique, Classic System Solutions, 101B Sand Creek Road - Suite 209 - Brentwood, CA 94513, 1995.
- [33] James HOBART : Designing successful mobile applications. Rapport technique, Classic System Solutions, 101B Sand Creek Road - Suite 209 - Brentwood, CA 94513, 2001.

- [34] Giulio IACUCCI, Antti JUUSTILA, Kari KUUTTI, Pekka PEHKONEN et Arto YLISAUKKO-OJA : *Connecting Remote Visits and Design Environment : User Needs and Prototypes for Achitecture Design*, volume 2795 de *Lecture Notes in Computer Science*, pages 45–60. Springer Berlin / Heidelberg, 2004.
- [35] Kenneth KENISTON : Software localization : Notes on technology and culture. <http://web.mit.edu/~kken/Public/papers1/Software%20localization.htm>, Janvier 1997.
- [36] A. KRALISCH, A.W. YEO et N. JALI : Linguistic and cultural differences in information categorization and their impact on website use. In *System Sciences, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, volume 5, pages 93b–93b, 04-07 Jan. 2006.
- [37] Bénédicte LE GRAND : Topic maps visualization.
- [38] Bénédicte LE GRAND : *Extraction d'information et visualisation de systèmes complexes sémantiquement structurés*. Thèse de doctorat, Université Pierre et Marie Curie, Paris VI, 2001.
- [39] Bénédicte LE GRAND et Michel SOTO : Topic maps, rdf graphs and ontologies visualization. Rapport technique, Laboratoire d'Informatique de Paris 6 (LIP6).
- [40] Eetu MAKELÄ, Eero HYVÖNEN et Samppa SAARELA : *Ontogator - A Semantic View-Based Search Engine Service for Web Applications*, volume 4273 de *Lecture Notes in Computer Science*, pages 847–860. Springer Berlin / Heidelberg, 2006.
- [41] Céline MARIAGE, Jean VANDERDONCKT, Abdo BEIREKDAR et Monique NOIRHOMME : DESTINE : outil d'aide à l'évaluation de l'ergonomie des sites web. In *IHM 2004 : Proceedings of the 16th conference on Association Francophone d'Interaction Homme-Machine*, pages 117–124, New York, NY, USA, 2004. ACM Press.
- [42] Natasa MILIC-FRAYLING, Ralph SOMMERER, Kerry RODDEN et Alan BLACKWELL : *SmartView and SearchMobil : Providing Overview and Detail in Handheld Browsing*, volume 2954 de *Lecture Notes in Computer Science*, pages 158–171. Springer Berlin / Heidelberg, 2004.
- [43] G. MOORE : Rdf and topic maps, an exercise in convergence. Berlin, 21-25 Mai 2001. XML Europe 2001.
- [44] Mikko NIKKANEN : *One-Handed Use as a Design Driver : Enabling Efficient Multi-channel Delivery of Mobile Applications*, volume 2954 de *Lecture Notes in Computer Science*, pages 28–41. Springer Berlin / Heidelberg, 2004.
- [45] Eyal OREN, Renaud DELBRU et Stefan DECKER : *Extending Faceted Navigation for RDF Data*, volume 4273 de *Lecture Notes in Computer Science*, pages 559–572. Springer Berlin / Heidelberg, 2006.
- [46] Gary PERLMAN : The firstsearch user interface architecture : Universal access for any user, in many languages, on any platform. Rapport technique, OCLC Online Computer Library Center, 2000.
- [47] Ben SHNEIDERMAN : The eyes have it : A task by data type taxonomy for information visualizations. In *IEEE Visual Languages*, numéro UMCP-CSD CS-TR-3665, pages 336–343, College Park, Maryland 20742, U.S.A., 1996.
- [48] Flavia SPARACINO, Glorianna DAVENPORT et Alex PENTLAND : City of news : cataloguing the world wide web through virtual architecture.
- [49] Lockheed Martin Space Missions SYSTEMS et SERVICES : Human research facility (hrf) human-computer interface (hci) design guide. Rapport LS-71130, National Aeronautics and Space Administration (NASA), Lyndson B. Johnson Space Center Houston, Texas 77058, Novembre 1997.
- [50] Sakari TAMMINEN, Antti OULASVIRTA, Kalle TOISKALLIO et Anu KANKAINEN : *Understanding Mobile Contexts*, volume 2795 de *Lecture Notes in Computer Science*, pages 17–31. Springer Berlin / Heidelberg, 2004.
- [51] Luc VAN LANCKER : Xml + xsl. Rapport technique, Rue des Brasseurs, 22 7700 Mouscron Belgium, 2001.

[52] W3C : Mobile device. Web.

[53] W3C : Web content accessibility guidelines 2.0. Publication web, [www.w3.org/TR/WCAG20/complete.html](http://www.w3.org/TR/WCAG20/complete.html), Avril 2006. W3C Working Draft.



---

Code source du layout du site web

---

## A.1 Onglet Description

### A.1.1 Code HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.
w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" media="screen" href="Layout3_2.css"
/>
<title>Titre / Nom — Fiche de description</title>
</head>

<body>
<div id="conteneur">
  <h1>
    Titre / Nom
  </h1>
  <div id="one">
    1
  </div>
  <div id="two">
    2
  </div>
  <div style="clear: both;">
  </div>
  <div id="photo">
  </div>
  <div id="desc">
    Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam
    nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam
```

```

        erat, sed diam voluptua. At vero eos et accusam et justo duo
        dolores et ea rebum. Stet clita kasd gubergren, no sea takimata
        sanctus est Lorem ipsum dolor sit amet.
</div>
<div id="panier">
</div>
<div id="menu">
<table width="240px" border="0">
    <tr>
        <td>PrÃsents</td>
        <td>Coord.</td>
        <td>Horaires</td>
        <td>DÃtails</td>
        <td>Ãquip. /<br />Services</td>
        <td>RÃpart. <br />Logement</td>
    </tr>
</table>
</div>
</div>
</body>
</html>

```

### A.1.2 Code CSS

```

*{
    margin:0;
    padding:0;
    font-family: Verdana;
    font-size:10px;
}

#conteneur{
    width:230px;
    height:310px;
    padding:5px;
    border:1px solid #000000;
}

h1{
    float:left;
    width:170px;
    height:40px;
    border:1px solid #000000;
    text-align:center;
    padding-top:15px;
}

#one{
    float:right;
    margin-top:7px;
    width:20px;
}

```

```

        height:17px;
        border:1px solid #000000;
        text-align:center;
        padding-top:3px;
    }

    #two{
        position:absolute;
        margin-top:41px;
        margin-left:188px;
        width:40px;
        height:17px;
        border:1px solid #000000;
        text-align:center;
        padding-top:3px;
    }

    #photo{
        float:left;
        width:172px;
        height:107px;
        background-image:url(photo3_2.jpg);
    }

    #panier{
        width:91px;
        height:24px;
        background-image:url(panier3_2.jpg);
        margin-top:10px;
        margin-left:auto;
        margin-right:auto;
    }

    #menu{
        position:fixed;
    }

    table{
        margin-top:6px;
        margin-left:-5px;
    }

    tr,td{
        font-size:8px;
        border:1px solid #000000;
    }

```

## A.2 Onglet Coordonnées

### A.2.1 Code HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.
w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" media="screen" href="Layout6.css" />
<title>Titre / Nom — Coordonnées</title>
</head>

<body>
<div id="conteneur">
  <h1>
    Titre / Nom<br />
    1ère partie de l'adresse<br />
    2ème partie de l'adresse<br />
     Téléphone / Fax<br />
     Email / Site Internet
  </h1>
  <div id="meteo">
    Météo
  </div>
  <div style="clear: both;">
  </div>
  <div id="photo">
    Google Maps
  </div>
  <div id="desc">
    Vous aimerez aussi :
    <div class="miniapercu1">
      Mini <br />
      aperçu
    </div>
    <div class="miniapercu3">
      Mini <br />
      aperçu
    </div>
    <div class="miniapercu2">
      Mini <br />
      aperçu
    </div>
    Et bien d'autres encore...
  </div>
  <div id="menu">
    <table width="240px" border="0">
      <tr>
        <td>Présent.</td>
        <td>Coord.</td>
        <td>Horaires</td>
      </tr>
    </table>
  </div>
</div>
</body>
</html>
```

```

<td>DÃtails</td>
<td>Ãquip. /<br />Services</td>
<td>RÃpart. <br />Logement</td>
</tr>
</table>
</div>
</div>
</body>
</html>

```

### A.2.2 Code CSS

Le code source du CSS est le mme que pour l'onglet Description dans l'ensemble. Il est disponible sur le CD-Rom de la version lectronique de ce mmoire.

## A.3 Choix du type d'objet dans une recherche

### A.3.1 Code HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.
w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" media="screen" href="Layout15.css" /
>
<title>Choix du type d'objet</title>
</head>

<body>
<div id="conteneur">
  <h1>
    Choix du type d'objet :
  </h1>
  <div id="desc">
    <form name="choixTypeObj" action="validate.php" method="post">
      <div class="miniapercu1">
        <br />
        <input type="radio" name="choix1" checked="checked" />
      </div>
      <div class="miniapercu3">
        <br />
        <input type="radio" name="choix1" />
      </div>
      <div class="miniapercu2">
        <br />
        <input type="radio" name="choix1" />
      </div>
    </form>
    <div id="precedent">
      << Précédent <<
    </div>
    <div id="suivant">
      >> Suivant >>
    </div>
  </div>
</body>
</html>
```

### A.3.2 Code CSS

Le code source du CSS est le même que pour la composition de la famille, sans les instructions concernant le composant « spinButton ». Il est disponible sur le CD-Rom de la version électronique de ce mémoire.

## A.4 Composition de la famille dans une recherche

### A.4.1 Code HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.
w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" media="screen" href="Layout16.css" /
>
<link type="text/css" rel="stylesheet" href="JQuerySpinBtn.css" />
<script type="text/javascript" src="jquery.js"></script>
<!--<script type="text/javascript" src="http://jquery.com/src/latest/"></
script-->
<script type="text/javascript" src="JQuerySpinBtn.js"></script>
<title>Composition de la famille</title>
<script type="text/javascript" language="javascript">
// Apply the SpinButton code to the appropriate INPUT elements:
$(function(){
    $("INPUT.spin-button").SpinButton({min:0});
});
</script>
</head>

<body>
<div id="conteneur">
    <h1>
        Composition de la famille :
    </h1>
    <div id="desc">
    <form name="choixTypeObj" action="validate.php" method="post">
    <table align="center">
        <tr>
            <td>
                
            </td>
            <td>
                <input type="text" class="spin-button" id="
                txtSpin" value="0" />
            </td>
        </tr>
        <tr>
            <td>
                
            </td>
            <td>
                <input type="text" class="spin-button" id="
                txtSpin" value="0" />
            </td>
        </tr>
    </table>
    </div>
    </div>
</body>
```

```

        <td>
            
        </td>
        <td>
            <input type="text" class="spin-button" id="
                txtSpin" value="1" />
        </td>
    </tr>
</table>
</div>
</form>
<div id="precedent">
    << Précédent <<
</div>
<div id="suivant">
    >> Suivant >>
</div>
</div>
</div>
</body>
</html>

```

#### A.4.2 Code CSS

```

*{
    margin:0;
    padding:0;
    font-family: Verdana;
    font-size:10px;
}

#conteneur{
    width:230px;
    height:310px;
    padding:5px;
    border:1px solid #000000;
}

h1{
    padding-top:50px;
    width:230px;
    height:30px;
    text-align:center;
}

#meteo{
    float:right;
    margin-top:7px;
    width:60px;
    height:37px;
    border:1px solid #000000;
}

```



```

        text-align:center;
        padding-top:33px;
    }

    #photo{
        width:228px;
        height:60px;
        border:1px solid #000000;
        text-align:center;
        padding-top:50px;
    }

    #desc{
        margin-top:5px;
    }
    #precedent{
        text-align:center;
        margin-top:20px;
        float:left;
        width:100px;
        height:15px;
        border:1px solid #000000;
    }
    #suivant{
        text-align:center;
        margin-top:20px;
        float:right;
        width:100px;
        height:15px;
        border:1px solid #000000;
    }

```

### A.4.3 Javascript

Le code source des javascript pour gérer le « spinButton » (incrémenter ou décrémenter une valeur numérique) et son contrôle a été trouvé sur le site de **SoftwareUnity** à l'adresse suivante : [www.softwareunity.com/sandbox/JQuerySpinBtn/](http://www.softwareunity.com/sandbox/JQuerySpinBtn/). Le code source est trop long pour l'inclure dans le mémoire mais vous pouvez le trouver sur le site de **SoftwareUnity**.



#### **B.1 Script XSL d’affichage d’une fiche touristique**

Étant donné la longueur du fichier, nous avons décidé de ne pas inclure le code source. Il se trouve sur le CD-Rom de la version électronique du manuel.

#### **B.2 Fichier XML d’une fiche touristique**

Pour les mêmes raisons que le script XSL, nous avons décidé de ne pas inclure le code source d’un fichier XML du site de la Nièvre.



---

Code source des exemples de fiches touristique

---

## C.1 Fiche d'une activité

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE objet SYSTEM "structure.dtd">
<objet>
  <activite soustype="culturelle">
    <info>
      <nom>Musée du Louvre</nom>
      <coordonnees>
        <adresse>
          <rue>Rue du Louvre</rue>
          <code_postal>75000</code_postal>
          <ville>Paris</ville>
          <pays>France</pays>
        </adresse>
        <telephone>
          <indic_tel>+33</indic_tel>
          <num_tel>(0) 40.20.50.50</num_tel>
        </telephone>
        <url>www.lelouvre.fr</url>
      </coordonnees>
      <description>Le musée du Louvre est le plus grand
        musée parisien par sa surface (160 106 m² dont
        58 470 consacrés aux expositions). Situé au cœu
        de la ville de Paris, entre la rive droite de
        la Seine et la rue de Rivoli, dans le 1er
        arrondissement, le bâtiment est un ancien palais
        royal. La statue équestre de Louis XIV
        constitue le point de départ de l'axe historique
        , mais le palais n'est pas aligné sur cet axe. C
        'est l'un des plus anciens musées et le
        troisième plus grand au monde. Le Louvre possède
```

```

        une longue histoire de conservation artistique
        et historique de la France, depuis les rois
        capétiens jusqu'à nos jours.</description>
<coordgeo>
    <longitude>44.3214N</longitude>
    <latitude>3.5423E</latitude>
</coordgeo>
<noteinteret>5</noteinteret>
<langueparlee>francais</langueparlee>
<langueparlee>anglais</langueparlee>
<langueparlee>espagnol</langueparlee>
<langueparlee>italien</langueparlee>
<paiement>espèces</paiement>
<paiement>Visa</paiement>
<paiement>Mastercard</paiement>
<periodeouverture>
    <datedebut>
        <jour>01</jour>
        <mois>01</mois>
        <annee>2006</annee>
    </datedebut>
    <datefin>
        <jour>31</jour>
        <mois>12</mois>
        <annee>2006</annee>
    </datefin>
    <ouverture>
        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>9</heure>
            <minute>00</minute>
        </heure_ouv_matin>
        <heure_ferm_PM>
            <heure>18</heure>
            <minute>30</minute>
        </heure_ferm_PM>
    </ouverture>
    <ouverture>
        <ouvert_bool>>false</ouvert_bool>
    </ouverture>
    <ouverture>
        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>9</heure>
            <minute>00</minute>
        </heure_ouv_matin>
        <heure_ferm_PM>
            <heure>18</heure>
            <minute>30</minute>
        </heure_ferm_PM>
    </ouverture>
</periodeouverture>

```

```

        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>9</heure>
            <minute>00</minute>
        </heure_ouv_matin>
        <heure_ferm_PM>
            <heure>18</heure>
            <minute>30</minute>
        </heure_ferm_PM>
    </ouverture>
    <ouverture>
        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>9</heure>
            <minute>00</minute>
        </heure_ouv_matin>
        <heure_ferm_PM>
            <heure>22</heure>
            <minute>00</minute>
        </heure_ferm_PM>
    </ouverture>
    <ouverture>
        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>9</heure>
            <minute>00</minute>
        </heure_ouv_matin>
        <heure_ferm_PM>
            <heure>18</heure>
            <minute>30</minute>
        </heure_ferm_PM>
    </ouverture>
    <ouverture>
        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>9</heure>
            <minute>00</minute>
        </heure_ouv_matin>
        <heure_ferm_PM>
            <heure>16</heure>
            <minute>30</minute>
        </heure_ferm_PM>
    </ouverture>
</periodeouverture>
</info>
    <public_vise>Collégiens</public_vise>
    <public_vise>Adulte</public_vise>
    <duree_activite>
        <heure>8</heure>
        <minute>00</minute>
    </duree_activite>
    <prix_activite>

```

```

                <libelle_age>Enfant</libelle_age>
                <valeur>Gratuit</valeur>
            </prix_activite>
            <prix_activite>
                <libelle_age>Etudiant</libelle_age>
                <valeur>Gratuit</valeur>
            </prix_activite>
            <prix_activite>
                <libelle_age>Adulte</libelle_age>
                <valeur>9,80 Å</valeur>
            </prix_activite>
        </activite>
    </objet>

```



## C.2 Fiche d'un restaurant

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE objet SYSTEM "structure.dtd">
<objet>
  <restaurant soustype="gastronomique">
    <info>
      <nom>Les écuries du Château</nom>
      <coordonnees>
        <adresse>
          <rue>Château de Dampierre</rue>
          <code_postal>78720</code_postal>
          <ville>Dampierre en Yvelines</ville>
          <pays>France</pays>
        </adresse>
        <telephone>
          <indic_tel>+33</indic_tel>
          <num_tel>(0) 1.30.52.52.99</num_tel>
        </telephone>
        <fax>
          <indic_tel>+33</indic_tel>
          <num_tel>(0) 1.30.52.59.90</num_tel>
        </fax>
        <email>contact@lesecuriesduchateau.com</email>
        <url>www.lesecuriesduchateau.com</url>
      </coordonnees>
      <description>Situé dans la sellerie du Château de
        Dampierre le Restaurant Gastronomique offre un
        cadre chaleureux et une vue privilégiée sur le
        parc et le Château de Dampierre.
        En saisons vous profiterez de notre terrasse le long d'une magnifique pièce
        d'eau bordée de marronniers centenaire.</description>
      <noteinteret>5</noteinteret>
      <langueparlee>francais</langueparlee>
      <paiement>Espèces</paiement>
      <paiement>Visa</paiement>
      <paiement>Mastercard</paiement>
      <paiement>American Express</paiement>
      <paiement>Chèque</paiement>
      <periodeouverture>
        <datedebut>
          <jour>01</jour>
          <mois>01</mois>
          <annee>2006</annee>
        </datedebut>
        <datefin>
          <jour>18</jour>
          <mois>012</mois>
        </datefin>
      </periodeouverture>
    </info>
  </restaurant>
</objet>
```

```

        <annee>2006</annee>
</datefin>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
</ouverture>
<ouverture>
    <ouvert_bool>>false</ouvert_bool>
</ouverture>
<ouverture>
    <ouvert_bool>>false</ouvert_bool>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
    <heure_ouv_PM>
        <heure>19</heure>
        <minute>00</minute>
    </heure_ouv_PM>
    <heure_ferm_PM>
        <heure>0</heure>
        <minute>00</minute>
    </heure_ferm_PM>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>

```

```

        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
    <heure_ouv_PM>
        <heure>19</heure>
        <minute>00</minute>
    </heure_ouv_PM>
    <heure_ferm_PM>
        <heure>1</heure>
        <minute>00</minute>
    </heure_ferm_PM>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>16</heure>
        <minute>30</minute>
    </heure_ferm_matin>
</ouverture>
</periodeouverture>
<periodeouverture>
    <datedebut>
        <jour>04</jour>
        <mois>03</mois>
        <annee>2006</annee>
    </datedebut>
    <datefin>
        <jour>29</jour>
        <mois>07</mois>
        <annee>2006</annee>
    </datefin>
    <ouverture>
        <ouvert_bool>true</ouvert_bool>
        <heure_ouv_matin>
            <heure>11</heure>
            <minute>30</minute>
        </heure_ouv_matin>
        <heure_ferm_matin>
            <heure>15</heure>
            <minute>30</minute>
        </heure_ferm_matin>
    </ouverture>
    <ouverture>
        <ouvert_bool>false</ouvert_bool>

```

```

</ouverture>
<ouverture>
    <ouvert_bool>false</ouvert_bool>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
    <heure_ouv_PM>
        <heure>19</heure>
        <minute>00</minute>
    </heure_ouv_PM>
    <heure_ferm_PM>
        <heure>0</heure>
        <minute>00</minute>
    </heure_ferm_PM>
</ouverture>
<ouverture>
    <ouvert_bool>true</ouvert_bool>
    <heure_ouv_matin>
        <heure>11</heure>
        <minute>30</minute>
    </heure_ouv_matin>
    <heure_ferm_matin>
        <heure>15</heure>
        <minute>30</minute>
    </heure_ferm_matin>
    <heure_ouv_PM>
        <heure>19</heure>
        <minute>00</minute>
    </heure_ouv_PM>
    <heure_ferm_PM>
        <heure>1</heure>
        <minute>00</minute>
    </heure_ferm_PM>

```

```

        </ouverture>
        <ouverture>
            <ouvert_bool>true</ouvert_bool>
            <heure_ouv_matin>
                <heure>11</heure>
                <minute>30</minute>
            </heure_ouv_matin>
            <heure_ferm_matin>
                <heure>16</heure>
                <minute>30</minute>
            </heure_ferm_matin>
        </ouverture>
    </periodeouverture>
</info>
<type_standing>Bon confort</type_standing>
<type_gastronomie>Française classique</type_gastronomie>
<prix>[26;45]</prix>
<jardin_bool>true</jardin_bool>
<terrasse_bool>true</terrasse_bool>
<menu_enfant_bool prix_menu_enfant="19">true</
    menu_enfant_bool>
<parking_bool>true</parking_bool>
<acces_hand_bool>false</acces_hand_bool>
</restaurant>
</objet>

```

### C.3 Fiche d'un hébergement

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE objet SYSTEM "structure.dtd">
<objet>
  <autres>
    <hebergement soustype="gite">
      <info>
        <nom>Gîte 832 – VILLAPOURCON</nom>
        <coordonnees>
          <adresse>
            <rue>LE BANCHOT</rue>
            <code_postal>58370</code_postal>
            <ville>VILLAPOURCON</ville>
            <pays>France</pays>
          </adresse>
          <telephone>
            <indic_tel>+33</indic_tel>
            <num_tel>(0) 3.86.59.90.67</num_tel>
          </telephone>
          <fax>
            <indic_tel>+33</indic_tel>
            <num_tel>(0) 3.86.59.14.22</num_tel>
          </fax>
        </coordonnees>
        <description>
          Jolie maison morvandelle
          bénéficiant d'une belle vue sur
          le Morvan comprenant au rez-de-
          chaussée un séjour avec coin
          cuisine salon (TV) une chambre
          (1 lit 2 pers) salle de bains wc
          . A l'étage: grande mezzanine 2
          ch (1 lit 2 pers 2 lits 1 pers 1
          lit bébé) salle d'eau wc.
          Chauff. élect. en sus. Terrasse
          salon de jardin. Poss. ménage:
          50E. A 10 km de Bibracte site
          gallo-romain archéologique
          européen du Mont-Beuvray. A 13
          km de St Honoré-les-Bains cure
          thermale Casino. A 22 km de
          Château-Chinon musées du
          Septennat et des Costumes. A 14
          km du Haut-Folin: randonnées. A
          13 km de Moulins-Engilbert:
          musée du Charolais.
        </description>
        <photo>url de la photo</photo>
      </hebergement>
    </autres>
  </objet>
```

```

        <langueparlee>Anglais</langueparlee>
        <langueparlee>Espagnol</langueparlee>
        <paiement>Chèque</paiement>
        <paiement>CB</paiement>
        <periodeouverture>
            <datedebut>
                <jour>01</jour>
                <mois>01</mois>
                <annee>2006</annee>
            </datedebut>
            <datefin>
                <jour>31</jour>
                <mois>12</mois>
                <annee>2006</annee>
            </datefin>
        </periodeouverture>
    </info>
    <type_standing>Confortable</type_standing>
    <distinction>3 épis</distinction>
    <type_chambre>
        <libelle>Chambre lit 2 personnes</libelle>
        <nombre>2</nombre>
    </type_chambre>
    <type_chambre>
        <libelle>Chambre 2 lits 1 personne + 1 lit
            de bébé</libelle>
        <nombre>1</nombre>
    </type_chambre>
    <prix>[340;?]</prix>
    <tv_satellite>
        <television_fournit_bool>true</
            television_fournit_bool>
    </tv_satellite>
    <piscine_bool>>false</piscine_bool>
    <internet_bool>>false</internet_bool>
    <parking_bool>true</parking_bool>
    <acces_hand_bool>>false</acces_hand_bool>
    <non_fumeur_bool>>false</non_fumeur_bool>
    <animaux_bool>>false</animaux_bool>
    <enfants_bool>true</enfants_bool>
    <nbr_place>6</nbr_place>
</hebergement>
<machine_laver_privée_bool>>false</machine_laver_privée_bool>
>
<lave_vaisselle_bool>>false</lave_vaisselle_bool>
<frigo_bool>true</frigo_bool>
<congelateur_bool>>false</congelateur_bool>
<jardin_bool>true</jardin_bool>
<jeux_enfants_bool>>false</jeux_enfants_bool>
<petit_dej_bool>>false</petit_dej_bool>
<demi_pension_bool>>false</demi_pension_bool>
<pension_compl_bool>>false</pension_compl_bool>

```

<div data-bbox="191 277 485 351" data-label="Text"><p>&lt;/objet&gt; &lt;/autres&gt;</p></div>
--